SUSE 10.2 » Apache » VMware » SugarCRM

LINUX FORMAT

Главное в мире Linux

Март 2007 № 3 (90)

LXF DVD внутри!

Knoppix 5.1.1

Плюс: Fedora 6 LIVE, NetBSD, FreeBSD, OpenBSD!

Читайте новости и... читайте **LinuxFormat** см. с. 6

Ваши проблемы с Linux *решены!*

» Советы » Оборудование » Установка
» Искусство поиска неисправностей с. 26

Который torrent?

BitTorrent занимает 50% пропускной способности Всемирной сети – мы расскажем, какой клиент следует использовать с.18

РАССМОТРИМ И ОЦЕНИМ:

SUSE 10.2, ASPLinux 11.2, SoftMaker Office, wxWidgets 2.8, VMware 6, Kerio Mail Server 6.3

Аутентификация сделанная людьми и для людей

OpenID изменит вашу жизнь к лучшему с. 46

Это, пожалуй, станет убойной возможностью Perl 6...
Дэмьен Конвей интервью с.34

Linux center
www.linuxcenter.ru

LINUX FORMAT
Главное в мире Linux

К Вашим услугам...

Тема этого номера – устранение неполадок, и мы решили спросить у Команды LXF: Какой лучший совет можно дать пользователю Linux?

Пол Хадсон
Закрыть программу, съевшую все циклы CPU можно с помощью ground pound. Но только, если вы – Марио.

Грэм Моррисон
Избавьтесь от внешних раздражителей – разведитесь, отправьте подальше детей и заложите дверь кирпичом.

Майк Сондерс
Хочешь из *X Window System* уйти? Ctrl+Alt+Backspace – и ты на верном пути!

Эфрейн Эрнандес-Мендоса
Просто позвоните Грэму Моррисону 01225442244 – и он с радостью поможет вам. А может, и нет.

Ребекка Смолли
Заведите побольше знакомств в Сети, ибо ни один из ваших друзей не будет понимать, о чем вы говорите.

Эндрью Грегори
Зачесав челку, вплетя в волосы нарцисс, спою грустную песню о том, что меня все ненавидят... А потом спрошу Майка.

Марк Бейн
Первое место в моем Топ-10 лучших советов по Linux занимает top. Она расскажет, что происходит с вашей системой.

Нейл Ботвик
Любую проблему можно решить правильным применением команды man, Google и большого молотка.

Энди Ченнел
Выйдите замуж за хакера. Моя жена явно выбрала меня за умение работать в Linux – только не говорите ей, что я ламер.

Ричард Драммонд
Отправляйтесь в зоопарк и смотрите на пингвинов, пока на вас не снизойдет озарение – или не заболит голова.

Миссия журнала

- Пропаганда свободного ПО в России
- Продвижение решений с открытым кодом в бизнес-сообществе
- Поддержка российского Open Source сообщества
- Организация трибуны для разработчиков свободного ПО
- Обратная связь между разработчиками и потребителями ПО

Сделаем это вместе!

Номер журнала, который вы держите в руках, посвящен устранению различных неполадок и решению разнообразных проблем – от установки нужного драйвера видеокарты до подключения к беспроводной сети. Будем надеяться, что эти советы помогут вам в вашей повседневной работе с Linux, но давайте остановимся на секундочку и подумаем: а откуда идут все эти проблемы?

Основополагающий принцип Open Source гласит: больше глаз – меньше ошибок. Казалось бы, за 15 лет существования Linux можно было отладить систему до блеска, но этого не произошло – почему? Герой мюзикла «В джазе только девушки» говорит: «Nobody is perfect», и FOSS – не исключение из этого правила. Закрытые драйвера видеокарт нельзя включить в свободные дистрибутивы, а значит, общения с «неофициальными» репозиториями пока не избежать. Но кое-что вы все же можете сделать – сообщить об обнаруженных проблемах разработчикам. Если вы потратили два часа, разбираясь с какой-то неисправностью – поделитесь этим знанием, заполнив за десять минут форму отчета об ошибке в Bugzilla. Открытое ПО предполагает участие, а ошибок становится меньше тогда, когда множество глаз не только смотрят, но и видят, не так ли?

LXF

Валентин Синицын » Главный редактор info@linuxformat.ru

Как с нами связаться

Письма для публикации: letters@linuxformat.ru
Подписка и предыдущие номера: subscribe@linuxformat.ru
Техническая поддержка: answers@linuxformat.ru
Проблемы с дисками: disks@linuxformat.ru
Общие вопросы: info@linuxformat.ru
Web-сайт: www.linuxformat.ru

› **Адрес редакции:** Россия, Санкт-Петербург, ул. Гончарная, 23, офис 54.
› **Телефон редакции:** (812) 717-00-37. Дополнительная информация на стр.120

LINUX FORMAT

Март 2007 № 3 (90)

Содержание

Весь номер – прямо как на ладони: приятного чтения!

Учебники

RPM
Легкая установка пакетов 54
Побаиваетесь устанавливать и удалять ПО? Узнайте, как это можно сделать, не прибегая к командной строке!

› **Autopackage: прост и понятен.**

SugarCRM
Установим! 58
Подумываете обзавестись или обновить CRM-систему? В новой серии учебников вы найдете все, что вам нужно

Mono
Подключаем библиотеки 62
C# и Mono позволяют написать поисковую систему меньше чем за час! Не верите? Читайте этот учебник!

Grub
Разберитесь с загрузчиком 66
Скучный черно-белый экран, который вы видите при старте системы, можно сделать повеселее – и безопаснее, если хватит духу взяться за Grub!

GTK+
Знакомство Glade 2 70
Визуальное проектирование интерфейсов в духе классического Gnome

Unix API
Консольные приложения 74
Думаете, что все возможности терминала ограничиваются функцией printf()? А вот и нет!

JavaEE
Java Server Pages 78
Попробуем отделить мух – от котлет, зерна – от плевел, логику – от дизайна.

PostgreSQL
Настраиваем сервер 82
В конфигурационных файлах можно найти множество опций – на какие из них следует обратить внимание в первую очередь?

LaTeX
Классы документов 88
В LaTeX можно создавать книги, статьи, письма, презентации, слайды... и все благодаря классам. Аспиранты – узнайте, как быстро оформить диссертацию!

Blender
Текстурирование и анимация 92
Статичные модели Blender можно оживить и раскрасить – но для этого, как и в реальном мире, придется понять законы физики

LXF DVD90

Майк вам покажет 110

Knoppix 5.1.1

Самый популярный Live-дистрибутив – теперь с трехмерным рабочим столом. Не смогли настроить Compiz? Решение перед вами!

Fedora 6 Live

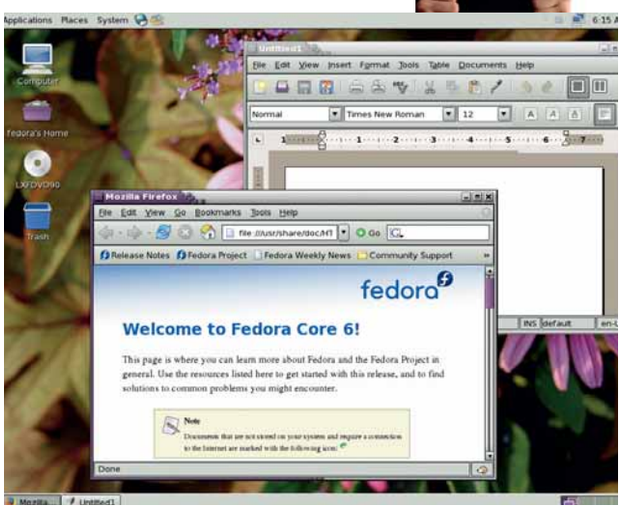
Возьмите community-дистрибутив от Red Hat с собой – теперь у него тоже есть Live-версия! Идеален для демонстрации возможностей Linux друзьям и коллегам.

NetBSD и OpenBSD

Надоел Linux? Откройте для себя новый мир свободных ОС с OpenBSD 4.0 и NetBSD 3.1

OpenOffice.org 2.1

Готовьтесь к празднику: новый OOo стал еще быстрее и лучше работает с форматами Microsoft!



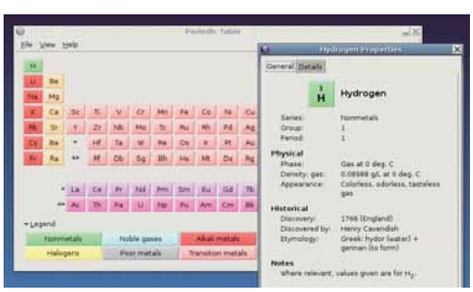
› **Fedora Core 6 в форме живого дистрибутива. Попробуйте, не устанавливая!**

Что за штука... OpenID?

Авторизация без лишних усилий
с. 46

LXFHotPicks

Лучшие новинки открытого ПО на планете 104



› **Так какой, говорите, ионный радиус у водорода?**

Содержание

Подпишись на Linux Format и сэкономь!

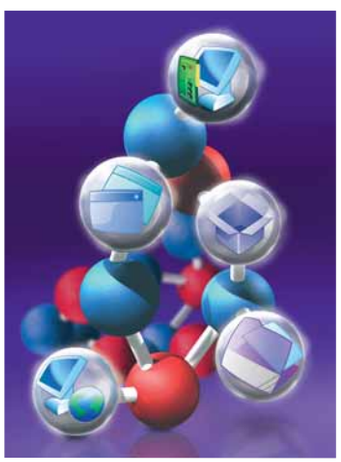
LXF DVD внутри!

См. страницу 110

Спецрепортаж

Проблемы Linux решены!

Узнайте, почему случаются неполадки и как с ними бороться на с. 26



А также...

Перо вам в шляпу! ..... с. 38
Модули Apache, которые стоит попробовать

Лингва эзотерика..... с. 42
По-настоящему странные языки программирования

Тяжелая артиллерия..... с. 48
Рабочие станции по обработке цифрового звука

Постоянные рубрики

Новости 04

DistroWatch 24
Ладислав Боднар изучает изменения в стане разработчиков Fedora. Посмотрим, на что будет похожа седьмая версия...

Интервью LXF 34
Еще больше энтузиазма по поводу Perl 6: узнайте, что заставляет двигаться Дэмьена Конвея.

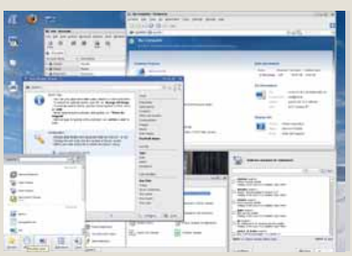
Что за штука 46
Новый способ аутентификации пользователей: OpenID.

Ответы 97
От разрешения экрана до настройки VNC-сервера – мы постараемся рассказать обо всем.

Через месяц .....
Что ждет вас в апреле

Обзоры

OpenSUSE 10.2 08
Если SUSE 10.1 заставил вас призадуматься о том, во что играет Novell – взгляните на SUSE 10.2: новое K-меню, интеграция Gnome и KDE плюс надежность.



› OpenSUSE 10.2 доказывает, что приложения GNOME прекрасно чувствуют себя и в KDE.

SoftMaker Office 2006 10
Компактный офисный пакет, предлагающий производительность за деньги. Мы сравнили его электронную таблицу с OOo Calc.

VMWare Workstation 6 12
Виртуализация – это круто и некоторым из нас всегда недостаточно предоставляемых возможностей. Удовлетворят ли их аппетиты 64-битность и возможность работы «в фоне»?

wxWidgets 2.8 13
Сообщество wxWidgets, как обычно, потрудилось на славу. Что нового в не слишком известном, но стоящем GUI-инструментарии?

ASPLinux 11.2..... 14
Отечественные разработчики не часто балуют нас регулярными релизами – но ASPLinux 11.2 заслуживает самого пристального внимания.

Kerio Mail Server 6.3..... 16
Коммерческий почтовый сервер типа «все в одном»: сумеет ли он потеснить разрозненные открытые альтернативы?

Сравнение: BitTorrent clients

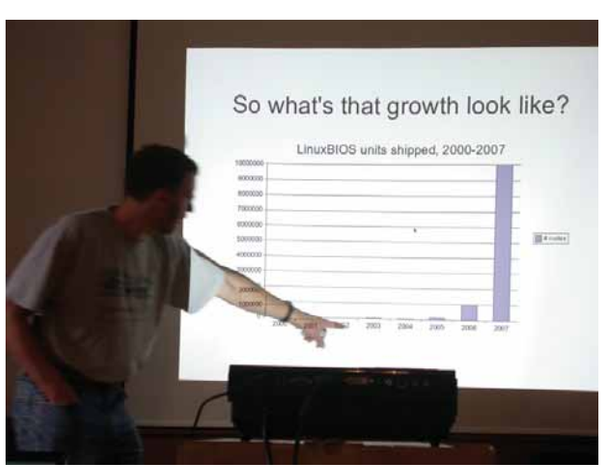
BitTorrent..... 19
Deluge 19
Azureus..... 20
BitTornado..... 21
KTorrent..... 21
TorrentFlux 22
Transmission..... 22

ГЛАВНЫЕ НОВОСТИ: » LinuxBIOS » Меняем ODF на OOXML » Linspire переходит на Ubuntu » Free Software по доверенности » Власть и бизнес России заинтересованы в свободном ПО

» Рубрику ведет Илья Шпаньков

LinuxBIOS – тернистый путь к полной свободе

В большинстве случаев любой компьютер, работающий под управлением свободной операционной системы, обладает как минимум одним проприетарным программным компонентом – BIOS. Впрочем, и для этого ПО есть свободная замена – уже несколько лет ведется разработка *LinuxBIOS*, способного полноценно заменить закрытый код и даже обеспечить лучшие показатели. Но, к сожалению, несмотря на явные успехи, к числу которых можно отнести выбор данного ПО для амбициозного проекта OLPC, разрабатывающего сверхдешевые ноутбуки для детей из развивающихся и экономически слабых стран, а также использование *LinuxBIOS* в индийских интернет-терминалах, турецких цифровых телевизорах и многих других специализированных устройствах, в общей сложности насчитывающих около 10 миллионов единиц, на пути в обычные компьютеры данный проект сталкивается с массой проблем. Вкратце сегодняшнюю ситуацию с разработкой *LinuxBIOS* обрисовал автор данного ПО – Рональд Миннич [Ronald G. Minnich] в интервью организаторам намеченной на конец февраля FOSDEM (Free and Open source Software Developers' European Meeting).



В частности, Рональд сказал, что такие компании, как SiS, VIA, Ali и Intel, поначалу активно поддержали его идею, но в последние годы практически перестали оказывать поддержку проекту. В то же время компании AMD, MSI, Linux Labs, iRobot и Linux Networx (являющаяся, кстати, ключевым партнером проекта), наоборот, активизировались и продолжают оказывать неоценимую помощь в создании и отладке свободной реализации BIOS. Подобная неравномерность отрицательно сказывается на разработке, которая по всем параметрам вполне может полноценно заменить проприетарный вариант. При этом основной камень преткновения – нежелание коммерческих компаний приоткрывать завесу тайны над аппаратными «секретами» своих устройств и отсутствие свободных драйверов. Между тем, опыт применения *LinuxBIOS* на бюджетных компьютерах проекта OLPC показывает, что использование свободного варианта сулит значительную выгоду: при использовании свободного аналога BIOS значительно снижается энергопотребление ноутбуков, а также ускоряется работа. По словам Рональда, возврат компьютера с *LinuxBIOS* из состояния «сна» происходит практически мгновенно.

Также автор проекта высказал мнение по поводу продвигаемого многими вендорами Extensible Firmware Interface (EFI). По его словам, основная функция данной разработки – стоять на страже интересов производителей аппаратных компонентов, помогая сохранять в секрете архитектуру и микропрограммную «начинку» чипсетов. По сути, EFI можно оценить, как попытку создания своеобразного «BIOS DRM», что может основательно помешать развитию открытых аналогов. Впрочем, Рональд не теряет оптимизма и считает, что *LinuxBIOS* сможет занять достойное место в мире программного обеспечения.

» Пока верстался номер

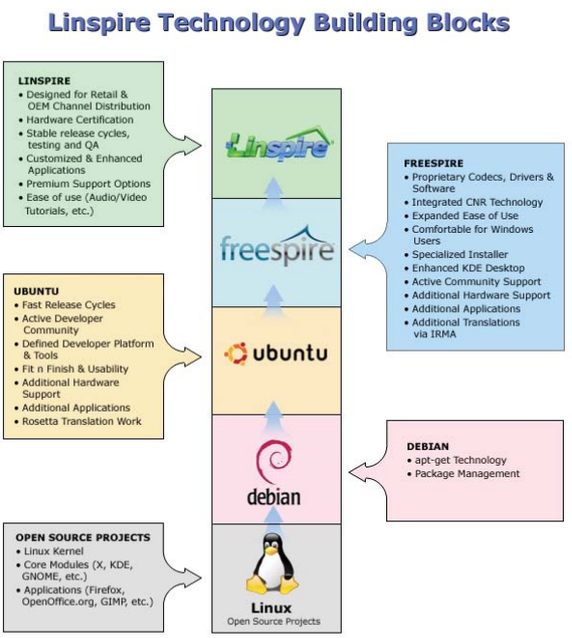
Компания GIGABYTE анонсировала выпуск материн-ской платы M57SLI-S4 с *LinuxBIOS* на борту. Ориентированная на процессоры AMD (сокет AM2) плата базируется на чипсете NVIDIA nForce 570 SLI, поддерживает до 16 ГБ памяти DDR-2 и имеет встроенный гигабитный Ethernet-контроллер.

Linspire, основанный на Ubuntu, основанном на Debian

В полне вероятно, что 2007 год станет наиболее щедрым на различного рода объединения и альянсы, что в будущем принесет немало интересных результатов. Впрочем, ближайший сюрприз уже не за горами: компания *Linspire*, занимающаяся разработкой одноименного коммерческого дистрибутива и его свободного аналога Freespire, заключила соглашение со спонсором проекта Ubuntu, компанией Canonical Ltd. По этому договору следующие версии *Linspire* будут основаны не на Debian, а на Ubuntu. Таким образом, *Linspire* планирует получить в свое распоряжение более свежие версии программ, традиционно присутствующих в Ubuntu, в отличие от консервативного Debian, который, в свою очередь, является базовым дистрибутивом для Ubuntu. Одновременно разработчики *Linspire* ожидают повышение интереса пользователей к своим программным продуктам благодаря хорошему рейтингу Ubuntu и более терпимому отношению его разработчиков к использованию коммерческого программного обеспечения.

Естественно, разработчики и пользователи дистрибутива, продвигаемого вторым в мире космотуристом Марком Шаттлвортом [Mark Shuttleworth], также не останутся в накладе и получат от *Linspire* интегрированный пакет CNR (Click and Run), упрощающий установку и обновление программного обеспечения, в том числе и коммерческого. Незадолго до этого компания *Linspire* опубликовала код CNR под свободной лицензией, и ожидаемый к весне Ubuntu-7.04 станет первым дистрибутивом, в который войдет данное решение. Одним из первых результатов соглашения стал отказ авторов Ubuntu от автоматической установки в дистрибутив несвободных видеодрайверов. Что, впрочем, не доставит проблем пользователям: благодаря CNR они легко смогут установить необходимое ПО, которое будет присутствовать на установочных дисках.

Многие известные деятели Open Source (включая Яна Мердока [Ian Murdock], основателя Debian) высказали явное одобрение по поводу такого поворота событий. По общему мнению, подобные виды сотрудничества принесут пользу как разработчикам дистрибутивов, так и всему сообществу в целом. На самом деле, в соглашении *Linspire* и Canonical есть один не очень приятный момент: в результате реализации договоренностей довольно большой объем закрытого ПО в виде проприетарных кодеков или видеодрайверов «просочится» в Linux-системы, традиционно старающиеся не связываться с несвободными продуктами. Но один из сооснователей OSI (Open Source Initiative) Эрик С. Реймонд [Eric S. Raymond] считает, что во имя популяризации Linux можно пойти и на такой рискованный шаг: увеличение рыночного присутствия операционных систем, основанных на GNU/Linux, по его мнению, является достаточно веской причиной, чтобы немного поступиться принципами. Кстати, сам Реймонд недавно вошел в состав правления *Linspire*.

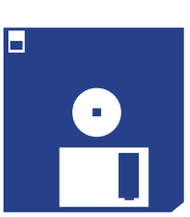


Linspire and Freespire build upon the work of other important open source projects and Linux distributions, with both Linspire and Freespire adding unique features for their intended audiences.

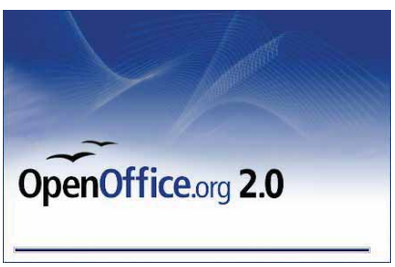
Власть и бизнес России заинтересованы в свободном ПО

9 февраля 2007 года состоялся круглый стол, организованный представителями Инфокоммуникационного Союза при поддержке Мининформсвязи России и Росинформтехнологии, темой которого стало обсуждение проблем разработки и использования свободного программного обеспечения в Российской Федерации. Участники встречи отметили, что роль свободного ПО в мире растет каждый год, а это говорит о наличии определенных преимуществ у Free Software по отношению к проприетарным продуктам. В этом ракурсе российским компаниям, разрабатывающим программное обеспечение, а также потребителям ПО следует более внимательно отнестись к свободным разработкам, особенно в плане возможной замены уже используемых проприетарных программных продуктов.

Между тем, в России продвижение свободного ПО связано с определенными трудностями, главными из которых участники встречи назвали отсутствие скоординированности, разрозненность материально-технической базы и отсутствие высококвалифицированных специалистов, способных обеспечить полноценное использование возможностей свободного ПО в бизнесе и государственном секторе. В связи с этим на встрече было озвучено предложение о создании на базе Инфокоммуникационного Союза координационного центра, в задачи которого войдет разработка государственной программы поддержки свободного ПО в России. Представители власти, присутствующие на встрече, одобрили инициативу бизнесменов и гарантировали обеспечить содействие новому проекту на уровне государственных властных структур.



Взаимовыгодный обмен форматами



Нашумевшее соглашение между компаниями Microsoft и Novell начинает приносить первые плоды. Вопреки ожиданиям скептиков, их можно отнести, скорее, к положительным результатам, позволяющим еще больше облегчить жизнь пользователей, работающих с документами, создаваемыми в пакетах офисных приложений *MS Office* и *OpenOffice.org*. В частности, в начале февраля спонсируемый компанией Microsoft проект по созданию конвертора документов представил первый релиз своего детища. Модуль позволяет работать с текстовыми файлами *MS Word* в двух форматах по выбору – или в *Office Open XML* (*OOXML*), являющимся форматом по умолчанию в *MS Office*, или в *OpenDocument* (ODF), недавно принятом в качестве международного стандарта. Разработчики гарантируют совместимость конвертора практически со всеми популярными версиями офисных пакетов редмондского гиганта, включая новейший *MS Office 2007*. Модуль распространяется под открытой лицензией BSD и доступен к загрузке на официальном сайте проекта. В дальнейшем, силами этих же разработчиков, планируется добавить поддержку ODF в другие офисные приложения – электронные таблицы *Microsoft Excel* и программу для создания презентаций *PowerPoint*. Завершить данную работу планируется к концу текущего года.

В свою очередь, компания Novell в декабре прошлого года внесла соответствующие коррективы в собственную редакцию пакета офисных приложений *OpenOffice.org*, добавив возможность работы с документами формата *OOXML*. Таким образом, вопрос несоответствия форматов двух популярных офисных пакетов можно считать практически закрытым, а всех пользователей поздравить с тем, что в работе с документами, создаваемыми в двух различных текстовых процессорах, стало одной проблемой меньше. Примечательно, что в дело «примирения» двух конкурирующих форматов документов внесли свою лепту и производители менее распространенных пакетов офисных приложений. В частности, компания *Corel* сообщила, что следующие версии пакета *WordPerfect Office* будут обладать полноценной поддержкой как ODF, так и *OOXML*.

Free Software по доверенности

Европейское отделение Free Software Foundation (FSFE) приступило к реализации нового проекта, в рамках которого создатели свободного программного обеспечения получают возможность делегировать авторские права на свои разработки любой организации или опекуну в соответствии с Доверительным Лицензионным Соглашением – Fiduciary Licence Agreement (FLA). Основное предназначение данного документа –- защита свободного ПО от возможных попыток нарушить авторские права на программный продукт в случаях, когда настоящий автор уже не существует, недоступен или не имеется возможности его установить. Таким образом FSFE надеется, что авторы программных продуктов смогут снять с себя массу проблем, связанных с решением возможных вопросов юридического характера и будут тратить все силы на совершенствование собственных разработок. При этом подобная «Доверенность» никоим образом не мешает лицензии GPL и не лишает создателей продукта авторских прав. Также планируется, что FLA позволит избежать случаев, когда коммерческие компании скупают свободные проекты, выпуская их затем по закрытой лицензии.

При всей полезности данного начинания, представители основного фонда FSF в лице Эбена Моглена [Eben Moglen] высказывают мнение, что FLA не сможет стать стопроцентной защитой свободного ПО. Особенно это касается случаев, когда разговор идет о различиях в законодательстве определенных стран. В частности, в США, где передача авторского права является обычным делом, FLA будет эффективно работать. К примеру, дочерние проекты GNU Project наподобие GCC делегируют права своей вышестоящей организации, при этом нет законов, способных воспрепятствовать данному юридическому действию. Но в таких странах, как Франция и Германия помимо авторского права существует и такое юридическое понятие, как «моральное право», которое согласно законодательству данных держав не может быть отчуждено или передано другому лицу. В этом случае, считает Моглен, FLA может оказаться недействительной. В связи с этим Моглен опасается одной вещи – что FSFE начнет продвигать FLA, как единственно возможное решение, хотя на самом деле оно таковым не является и требует индивидуального рассмотрения в каждом конкретном случае.

http://www.fsfeurope.org/projects/fla/. lxf

Новости короткой строкой

» Компания Sun Microsystems планирует перевести свои проекты Java и Solaris на лицензию GPLv3 и создать операционную систему GNU на ядре Solaris.

» Компания Trolltech в сотрудничестве с VirtualLogix на Конгрессе 3GSM в Барселоне представили однопроцессорный телефон, управляемый операционной системой Linux в тандеме с системой виртуализации в режиме реального времени.

» Red Hat подписала соглашение с Вооруженными Силами Швеции, согласно которому военные заменят серверные системы Windows на аналогичные программные продукты от Red Hat.

» Очередная версия браузера *Netscape 9* будет выпущена одновременно для трех платформ – Windows, Linux и Mac OSX.

» В интернете открыт новый ресурс *MobileLinuxInfo.com*, предназначенный для пользователей, разработчиков мобильных телефонов и программного обеспечения на базе Linux.

» Министр образования Пермского края Николай Карпушин заявил, что для снижения вероятности нарушения действующего законодательства в учебных заведениях будет сведено к минимуму использование коммерческого ПО, а в первую очередь будет рассматриваться возможность использования свободных программных продуктов с открытым исходным кодом.

» Идя навстречу пожеланиям читателей, редакция журнала Linux Format опубликовала на web-сайте архив номеров за 2006 год в формате PDF.

LINUX FORMAT

Обзоры

Новинки программного и аппаратного обеспечения в описании наших экспертов

Алексей Федорчук
Свою первую (и последнюю) программу написал еще на Алголе.

Debian или Kebian?

На протяжении последнего времени в тематической печати наиболее часто фигурировали Red Hat и Debian со своими клонами. И как-то в тени остался третий кит Linux-дистрибуции, Slackware. Хотя исторически его следовало бы назвать первым.

Тем не менее, и он не стоит на месте, не смотря на всем известные осложняющие обстоятельства. И, что немаловажно, интенсивно развиваются его прямые потомки, в том числе и весьма юного возраста.

В их числе следует назвать в первую голову ZenWalk (http://www.zenwalk.ru) - дистрибутив, избравший своим тотемом дельфина. И развивающийся со стремительностью, присущей этому морзверю: в конце февраля вышла его очередная версия (4.4 – напомню, что предыдущие чередовались с интервалом менее чем в полгода).

Чем интересен этот дистрибутив? Если Ubuntu во всех его проявлениях можно считать одним из способов легкой, для начинающего пользователя, установки Debian, то ZenWalk играет ту же роль в отношении Slackware. Скачал образ диска размером несколько более 400 МБ, пользователь в считанные минуты получает компактную, но полностью готовую к употреблению систему – с чрезвычайно элегантно оформленным XFce в качестве рабочего, набором утилит, вполне достаточным для счастья, легкими офисными пакетами и инструментами web-редактирования, не вполне полной, но достаточной для начала поддержкой русского языка.

А дальше для наращивания мощи перед ним два пути. Первый – наименьшего сопротивления, то есть использование встроенной системы пакетного менеджмента, netpkg, не более сложной в обращении, чем apt или pacman. Второй же – традиционное для Slackware и его потомков конструирование собственной системы.

alv@posix.ru

Сегодня мы рассматриваем...

08 OpenSUSE 10.2
Едва расписавшись с Microsoft, SUSE выпустила новый дистрибутив. Стал ли он на 0.1 лучше, чем 10.1, или же Novell все эти месяцы занималась чем-то другим? **8/10**

10 SoftMaker Office 2006
Нас вряд ли назвать можно гениями литературы, но мы удостоили Softmaker Office самых теплых слов. И вы их сейчас читаете! **7/10**

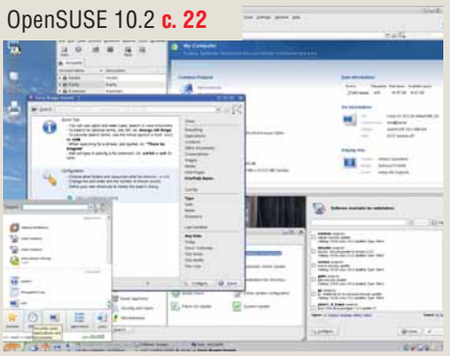
12 VMWare Workstation 6.0
С конвейера лидера в области виртуализационных технологий сошла новая модель – Workstation 6: наши первые впечатления. **8/10**

13 wxWidgets 2.8
Многие считают, что выбор GUI-инструментария сводится к выбору между GTK и Qt. wxWidgets демонстрирует, что «третий не лишний». **9/10**

14 ASPLinux 11.2
Отечественный настольный дистрибутив – глазами простого пользователя. Не SUSE единым... .... **8/10**

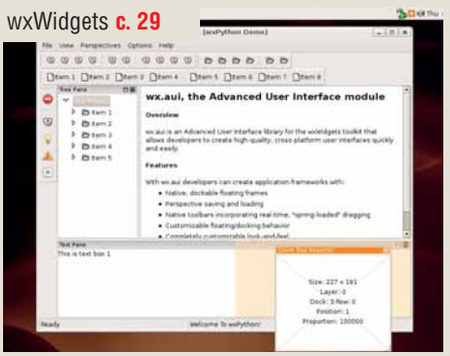
16 Kerio Mail Server 6.3
Есть ли жизнь без Sendmail и Postfix? Посмотрите на этот почтовый сервер-комбайн **6/10**

OpenSUSE 10.2 **с. 22**



› ***Beagle*! *F-Spot*! *Tomboy*! И, конечно, *Yast*! Нет, не может быть, чтобы все было совсем уж гладко...**

wxWidgets **с. 29**



› **Создание продвинутых пользовательских интерфейсов при минимуме кодирования.**

НАШ ВЕРДИКТ: пояснение

Все попавшие в обзор продукты оцениваются по одиннадцатибальной шкале (10 – высшая оценка, 0 – низшая). Как правило, мы оцениваем функциональность, производительность, простоту использования и цену, а для бесплатных программ учитывается документация. Кроме того, мы всегда выставляем общую оценку, демонстрирующую наше отношение к продукту.

Выдающиеся решения могут получить престижную награду «Top Stuff». Номинантами становятся лучшие из лучших – просто высокой оценки здесь недостаточно.

Рассматривая свободное ПО, мы обычно указываем предпочтительный дистрибутив. Иногда это означает компиляцию из исходных текстов, но, если разработчики рекомендуют Autopackage, мы следуем этому совету.

LINUX FORMAT TOP STUFF

LINUX FORMAT Вердикт

Google Earth

Разработчик: Google
Сайт: http://earth.google.com
Цена: Бесплатно по закрытой лицензии

Функциональность	10/10
Производительность	9/10
Простота использования	9/10
Документация	9/10

» *Если весь мир – сцена, то Google Earth – театр. Простая в использовании, захватывающая и ободряюще практичная программа.*

Рейтинг 9/10

OpenSUSE 10.2

Последнее время Novell находится в гуще всевозможных событий – **Нейл Ботвик** оценивает, а событие ли ее новый дистрибутив?

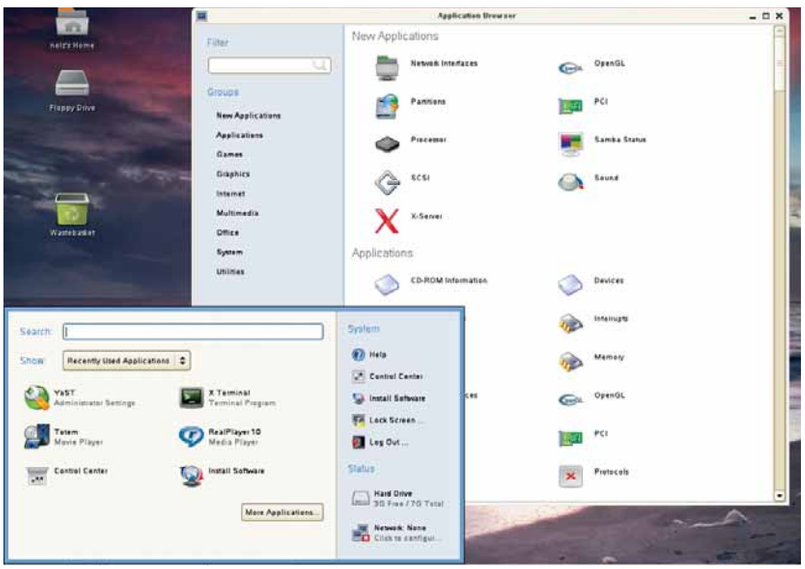
Вкратце...

» Linux-дистрибутив, спонсируемый (и в основном разрабатываемый) Novell, рассчитанный как на домашних, так и на бизнес-пользователей. См. также: Fedora Core 6, Mandriva 2007 или Ubuntu 6.10.

Недавно SUSE перенесла несколько ударов. Проблемы с OpenSUSE 10.1, особенно в области пакетного менеджмента и обновления… Сделка Novell с Microsoft, вызвавшие штурм Замка Novell разъяренными толпами воинов Свободного ПО (*см. раздел новостей* **LXF86**)… Проблемы 10.1 удивили: такое обнаруживают еще на стадии .0, а к .1 уже исправляют. Встает вопрос: смогли ли Novell и Сообщество SUSE исправить положение в OpenSUSE 10.2?

При инсталляции предлагается несколько рабочих столов – KDE, Gnome и «Другие» (Others) – но выбрать можно лишь один. Для пользователя-одиночки это нормально, но если у вашей лучшей половины вкус не совсем такой, как ваш, придется потом добавлять рабочий стол через *Yast*. Безопасность надежная, с брандмауэром *Yast* (установлен по умолчанию), традиционным iptables и защитой от вредных приложений *AppArmor*.

В инсталляторе есть возможность удаления IPv6 – некоторые распространенные модемы/роутеры не умеют с ним обращаться, а удаление может улучшить существующие подключения и даже восстановить разрушенные. Аналогично другим дистрибутивам, SUSE предлагает загрузить обновления программ во время установки. Разница в том, что система SUSE загружает лишь заплатки, а не пакеты целиком, что существенно снижает трафик, особенно для крупных пакетов.



› **Меню Gnome на SUSE может показать и часть приложений, и все сразу – выбор за вами!**

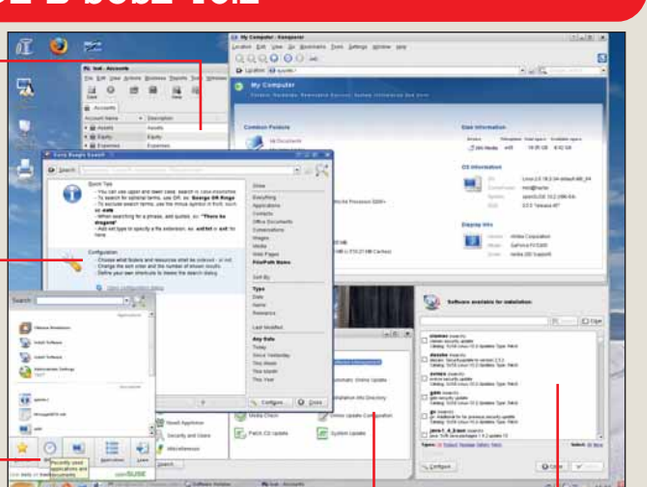
Как поступают теперь многие Linux-дистрибутивы, SUSE четко разграничивает свободу и функциональность: к пяти дискам можно добавить шестой, с несвободным ПО (*Adobe Reader*, Java и *Opera*). Имеется и соответствующее онлайн-хранилище для обновлений, напоминающее репозиторий Ubuntu Restricted. Только хранятся там в основном приложения, а не драйверы. OpenSUSE доступен и в виде DVD ISO, но учтите, что туда помещено все ПО, включая несвободное.

Время вбрасывания

SUSE заменил K-меню в KDE новым, «ориентированным на эффективность» меню под названием Kickoff (Вбрасывание). Меню обманчиво простое: по-прежнему доступно все, но ваши любимые приложения еще доступнее. Список избранного настолько удобен, что мы с него не слезали. В Kickoff встроен *Kerry*, KDE-оболочка поискового движка Beagle, делающая поиск вездесущим (в панели инструментов *Konqueror* тоже есть поисковый модуль *Kerry*, специально для нас, разгильдяев). Приложения Gnome на SUSE хорошо вписываются в KDE – *GTK*-программы, помещенные в KDE, выглядят как KDE-программы, и нет этих жутких визуальных искажений, получающихся при попытке запуска приложений от разных сред. Но KDE-приложения в Gnome по-прежнему выглядят ужасно.

Кое-кого расстроит удаление из ядра поддержки smbfs. Эта файловая система используется для монтирования общих с Windows ресурсов. Сама по себе возможность не исчезла, так как в ядре имеется CIFS, усовершенст-

Рабочий стол KDE в SUSE 10.2



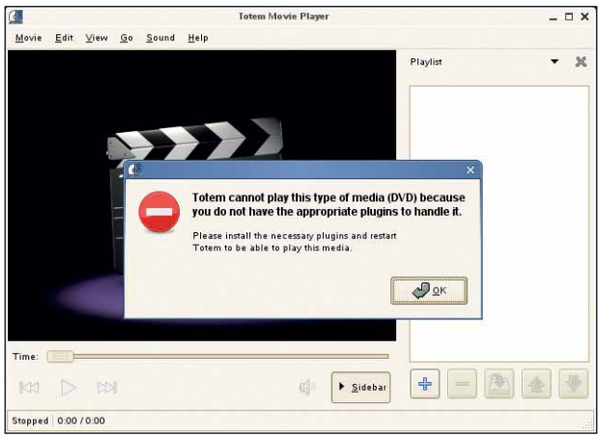
GTK на KDE
Аккуратная работа с темами приводит к тому, что *GTK*-приложения (здесь, *GnuCash*) чувствуют себя в KDE как дома.

Встроенный поиск
Kerry, KDE-оболочка для *Beagle*, поискового движка от Novell.

Kickoff
К новому меню SUSE для KDE (доступно и старое) нужно привыкнуть, но это несомненный прогресс.

Yast
Независимо от вашего отношения к идее конфигурировать все на свете одной программой, *Yast* упрощает основную работу по настройке.

ZENworks
Новый инсталлятор, отвлекающий ваше внимание от *Yast*. Не разумнее ли предоставить управление ПО единственному менеджеру?



› SUSE 10.2 не хочет воспроизводить DVD. Не только коммерческие, обычные тоже.

вованная замена. SUSE объясняет ликвидацию SMB тем, что он больше не поддерживается, но это лишь теория, поскольку с этом самом комплекте 10.2 есть приложения, предназначенные для работы с инструментами smbfs и рассчитанные на то, что соответст-вующий модуль имеется. Похоже, задумано было неплохо, но до конца дело не довели. Впрочем, с главной файловой системой все хорошо – это по-прежнему ext3 [раньше по умолчанию предлагался reiserfs, – прим. ред.].

Обновления

Система обновлений SUSE работает через менеджер пакетов *ZENworks*, и это странно: существуют две независимые системы управления пакетами. Один из вариантов – традиционный, через секцию управления пакетами *Yast*, а другой – новый, с помощью *ZENworks*. Оба распоряжаются одними и теми же RPM и базой данных, и можно переключаться между ними. Оба имеют свой список инсталляционных ресурсов, но они не синхронизированы – если ресурс добавлен в *ZENworks*, то при работе с *Yast* операцию придется повторить. Раздражает постоянный запуск синхронизации с *ZENworks*, сколько его ни отключай: это тормозит добавление пакетов. Обе системы обновления грешат отсутствием обратной связи. Часто невозможно понять, зависло приложение или просто загружает особо крупный пакет.

Грэм считает...

«Когда-то мне нравился OpenSUSE. Но теперь в нем отсутствует главное, чего я ожидаю от дистрибутива Linux: преданность Linux-сообществу».

Novell, MS и OpenOffice.org

В OpenSUSE 10.2 Novell включила «существенно улучшенную» версию *OpenOffice.org*. Некоторые улучшения выглядят полезными, например, интеграция с *Evolution* и *Kontact*, но одно из них вызвало бурю дебатов. Спорная функция – «улучшенная совместимость с файлами Microsoft Office», включающая поддержку макросов Visual Basic для Excel в Calc. Неудивительно, что в нее вцепились участники бойкота Novell. Мы протестировали работу макросов VB с помощью двух разных таблиц, и ничего не сработало. То ли Novell втихую убрала поддержку макросов, то ли эта поддержка не так хороша, как было заявлено.

Графические причуды

Владельцев видеокарты Nvidia ждет разочарование: для нее предусмотрены лишь свободные драйверы, а значит, не будет 3D-ускорения. SUSE не включает проприетарные драйверы Nvidia (даже в «несвободный» CD) по лицензионным причинам, но их несложно добавить через тот же *Yast* или *ZENworks*. На SUSE 10.2 можно установить *Xgl*, хотя это будет посложнее, чем, скажем, с Mandriva 2007. Нужно использовать редактор **/etc/sysconfig** в *Yast*, чтобы сменить X-сервер с *X.org* на *Xgl* и установить *Compiz*. Правда, настройка делается раз и навсегда, и в Gnome все отлично работает. К несчастью, в KDE доступны лишь немногие функции – никаких пульсирующих окон – но это скорее вопрос к *Compiz*, чем к SUSE.

Воспроизведение DVD на SUSE традиционно хромает. Согласны, защищенные диски проигрывать нельзя, и libdvdcss осталась за бортом. Но SUSE не в силах воспроизвести даже обычные DVD, созданные *Dvdauthor*. Необходима замена сломанного пакета xine-lib, что означает добавку еще одного репозитария к послеустановочной настройке *Yast*. MP3 воспроизводит *RealPlayer* – по крайней мере при инсталляции с «несвободного» диска или DVD – и здесь все в порядке.

Проблемы с мультимедиа вызваны скорее щекотливой юридической ситуацией, чем виной разработчиков. Не стоит обвинять Novell в чрезмерной осторожности, к тому же на месте, как обычно, Jem Report с подробными указаниями по добыче недостающих фрагментов, на **www.softwareinreview.com/cms/content/view/60**.

Если исключить вышеупомянутые мультимедиа-элементы, SUSE содержит поразительно широкий спектр ПО. Прошли те времена, когда в крупных дистрибутивах можно было установить пару рабочих столов и дюжину программ для каждой мелочи; выбор на DVD по-прежнему широк, но стандартная инсталляция проста и понятна.

Продуманный перечень программ и новые меню SUSE помогают быстро подобрать ПО для работы, но если вы желаете, чтоб из традиционного K-меню вываливалось 17 текстовых редакторов – пожалуйста. В целом набор ПО свежий: ядро 2.6.18, KDE 3.5.5, Gnome 2.16, *Firefox 2.0* и *Gimp 2.2.13*. Разработчики не успели включить новейшую версию *OpenOffice.org 2.1*, зато есть 2.0.4. Удивительно, что в 10.2 нет собственных разработок Novell – Mono 1.1.8 и *Beagle 0.2.12* на диске есть, но ведь 1.2.1 и 0.2.13 вышли на добрую неделю раньше сборки пакетов для диска. Пауза между выходом проекта и сборкой дистрибутива вполне естественна, но продукты-то разрабатывались в одной фирме!

Несмотря на некоторые нестыковки в управлении пакетами, SUSE 10.2 значительно лучше 10.1. В прошлый раз Novell споткнулась, но выправилась и произвела профессиональный, удобный в использовании (правда, слегка сыроватый) продукт. LXF

OpenSUSE или SLED?

Сегодня SUSE существует в двух ипостасях – OpenSUSE 10.2 и SUSE Linux Enterprise Desktop (SLED) 10.0 – какую же выбрать? На первый взгляд, Open SUSE лучше: она новее, и набор ПО более свеж. Но вы можете предпочесть уже опробованную, надежную вещь. Этим два SUSE и различаются. У OpenSUSE 10.1 были крупные проблемы, особенно по части управления пакетами, а SLED ультра-надежен.

Честно говоря, вы всегда рискуете, пробуя новейшую версию чего-нибудь, уже потому, что немногие этим чем-нибудь пользовались. SLED предназначен для бизнес-потребителей, для которых стабильность превыше новизны, особенно в тех случаях, когда система устанавливается на несколько сотен компьютеров. Для индивидуального пользователя это вопрос личного выбора – как всегда на Linux.

LINUX FORMAT Вердикт

OpenSUSE 10.2

Разработчик: Novell
Сайт: http://en.opensuse.org
Цена: Бесплатно, в основном под GPL

Функциональность	8/10
Производительность	7/10
Простота использования	9/10
Документация	7/10

» Разработчики SUSE оставили позади 10.1 и произвели солидный профессиональный дистрибутив, определенно заслуживающий инсталляции.

Рейтинг 8/10

SoftMaker Office

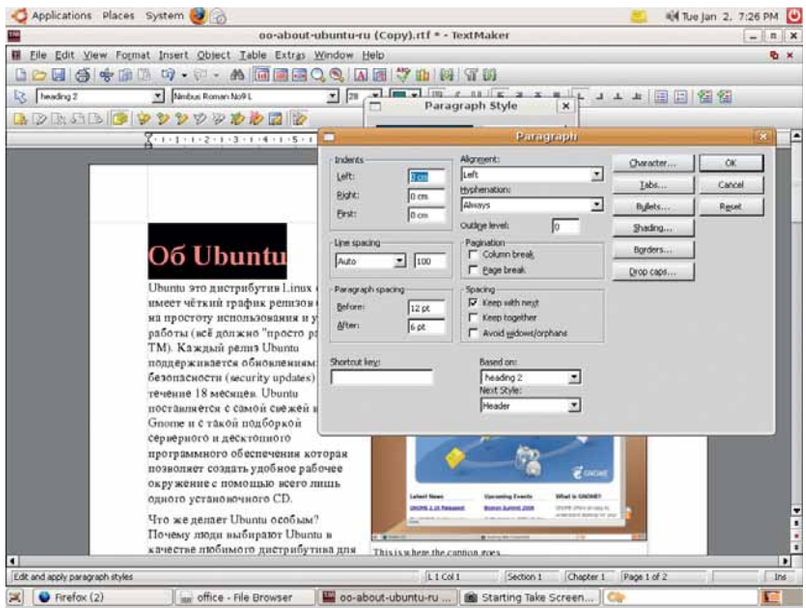
Конкурент *OpenOffice.org* – да еще и платный? SoftMaker решил, что это реально; осталось убедить **Энди Ченнела**…

Вкратце...

» Сцепка текстового процессора и базы данных для x86 Linux-дистрибутивов и FreeBSD. Попробуйте также: *OpenOffice. org*, *KOffice* или *AbiWord* совместно с Gnumeric.

Печать осталась в прошлом веке; теперь у нас есть txt, eml и SMS. Но в бизнесе люди все еще цепляются за бумажную копию, как за последнего пушистого котенка в зоомагазине. Для этих типов офисный комплект – центр вселенной, и выбор его необычайно важен. Пользователи Windows сгрудились вокруг *MS Office*, большинство линуксоидов счастливо с *OpenOffice. org*; а вот SoftMaker (г. Нюрнберг) вспахивает собственную ниву, производя комбинацию *TextMaker* и *PlanMaker* для Windows, Linux, Pocket PC и Windows CE (поддержки Mac нет).

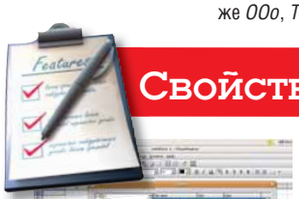
Пакет состоит из двух частей: текстового процессора и редактора электронных таблиц. Модуль создания презентаций отсутствует, и это минус *SoftMaker 2006* еще до «здрасте». Да, можно создать документ в *TextMaker*, а затем перевести его в формат HTML или PDF, но для профессиональных презентаторов этого мало. Нет и редактора иллюстраций, чем может похвалиться *OOo*, хотя в *TextMaker* имеются неплохие инструменты для иллюстрирования.



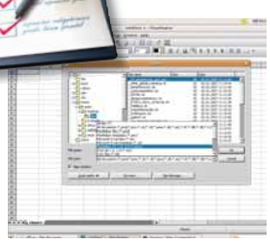
› **TextMaker имеет многоязыковую поддержку и удобные стили абзацев.**

> «Нестандартные виджеты делают его приятнее OOo на вид.»

Скудость комплекта неуместна еще и потому, что, в отличие от других офисных пакетов Linux, *SoftMaker 2006* стоит отнюдь не гроши. Будучи на несколько сотен фунтов дешевле MS Office, он в то же время почти на $70 дороже *OOo*, *ThinkFree* или *KOffice*. За такую сумму пакет должен реально превосходить аналоги, иначе кто ж его купит – а без презентаций и иллюстраций его преимущества нужно еще поискать. Поищем!



Свойства навскидку



Электронные таблицы
XLS, SDC, DBF… *PlanMaker* поддерживает широкий спектр форматов – и поддерживает надежно.

Форматирование
TextArt – ответ пакета на вордовский *WordArt* по стилям и расцветке текста. Работа та же, а цена меньше.

Легкий на ногу

Сначала об инсталляции. Там, где пожиратель памяти *OOo* словно шлепает по цифровому болоту, *SoftMaker Office*, умещаясь в скромные 32 МБ, проносится со скоростью велосипедиста. На нашем Intel Core 2 Duo *TextMaker* открылся и был готов к работе в 4 секунды – сравните с томительными 23 секундами *OOo* Writer – а *PlanMaker* развернулся за 5 (20 у *OOo Calc*).

Сам бинарник весит 23 МБ (*OOo* – 100 МБ с гаком), плюс 12 МБ интернациональных словарей. Комплект включает два статических двоичных файла, и как только они загружены, запуск сводится к распаковке архива и щелчку на одном из значков. То есть офис будет работать на любом Linux-дистрибутиве. Недостаток – нет тесной интеграции с вашим рабочим столом: файловые ассоциации и пункты меню нужно добавлять вручную. Для одиночного пользователя не проблема, но для группы компьютеров уже затруднительно.

Работа с текстом

Большинство пользователей, вероятно, найдет в *TextMaker* все, что им надо. Этот текстовый процессор имеет обычный набор инструментов плюс полезнейшие дополнения в виде буквиц, стилей символов и оформления абзацев, шаблонов страниц и свободных границ текстовых областей и рисунков.

Умиляет функция «живого» обтекания [wrap], позволяющая запросто перемещать границы текстовой области мышью, как в *Adobe InDesign*. Кроме базовых текстовых функций, *TextMaker* работает с DOC-совместимыми комментариями и колонтитулами и имеет зачаточные инструменты редактирования растровой графики.

В комплекте нет базы данных, поэтому полезно, что *TextMaker* предусмотрел соответствующий апплет для работы с почтой, файлами форматов DBF, CSV или собственного формата. Гибкость не совсем та, что у отдельного приложения, но для повседневной работы нормально.

Чего здесь действительно не хватает, так это вертикальной линейки в окне редактирования, то есть нельзя перемещать вертикальную направляющую, в отличие от горизонтальных: их надо добавлять через диалог. Не смертельно, но нудно, и, главное, непонятно, почему такого нет.

TextMaker – надежный, современный и удобный текстовый процессор. Органы управления у него нестандартные настолько, чтобы порадовать глаз, но не настолько, чтобы не понять, скажем, значение значка «Полужирный» (“Bold”). Внешность приятнее,

чем у *OOo Writer*, но достигается это ценой отсутствия интеграции с вашим рабочим столом и сглаженных шрифтов на панелях инструментов и в меню. Поддержка DOC (то есть импорт и экспорт) превосходна, вдобавок *TextMaker* умеет читать и записывать файлы и старых (SWX), и новых (ODT) форматов *OOo*, в дополнение к обязательным RTF и простому тексту. Вывод на печать также хорош. Приложение вполне стабильно, но когда мы все-таки умудрились его «завалить», программа исчезла без всякого предупреждения.

Пол считает...

«Безупречная совместимость с Excel обязательна для многих потенциальных мигрантов на Linux, и SoftMaker Office здесь превосходит OOo».

Как там с Excel...

PlanMaker – табличная часть SoftMaker, предназначенная для работы с более-менее сложными вычислениями. Как и текстовый напарник, имеет солидный набор функций и знакомый пользовательский интерфейс – без побрякушек – включающий многотабличную вкладочную организацию, крупную, дружественную панель формул и разнообразные функции форматирования ячеек. Панели инструментов разумно организованы и не переполнены параметрами, поэтому рабочее пространство не столь перегружено, как в *OOo Calc*. Основной расчет – на повседневные нужды среднего пользователя. А если ваш уровень выше (или ниже!) среднего, панели инструментов легко настроить через меню «Preferences» (Параметры). Редактор формул *PlanMaker* содержит широкий (хотя и похуже, чем у *Calc*) набор параметров и удобное управление операторами, делающие построение вычислений наглядным.

Разработчики проявили недюжинные способности в сфере работы с файлами *Excel*. Программа смогла открыть все, что было предложено (100 документов, вынутых из Сети в случайном порядке) почти без запинки, и в этой области она значительно превосходит *Calc* (*см. врезку «PlanMaker против Calc»* ниже). Мы заметили единственную проблему: случайные искажения встроенных рисунков во время навигации между таблицами.

Нам понравилась работа *PlanMaker* – интуитивно понятная и аккуратная. Поддержка макросов *Excel* надежна и стабильна, что для многих будет важнейшим качеством. Как мы ни старались, программа ни разу не только не зависла, но даже ощутимо не сбавила скорости обработки самых сложных таблиц. По сравнению с *Calc*, она не только быстрее запускается, но и быстрее переключается между разными таблицами и быстрее обновляет ячейки.

Для тех, кто сжился со своим *Excel* или накопил крупный архив, этот пакет может стать наилучшим вариантом обретения под Linux некоего Единого Офиса или даже самого *Excel*. Каждую переменную и макрос не протестируешь, но все предложенные документы *PlanMaker* обработал превосходно.

Если вы подумываете о миграции, учтите, что эта парочка – не клоны продукции MS: мы бы расценили их как нечто среднее между *OOo* и визуальным излишеством *MS Office 2003*. Фоновые градиенты не отвлекают от работы, но достаточно четко разграничивают панели инструментов, а самое замечательное, что это не ухудшает производительности программ. Чтобы упростить изучение, пакет предусматривает подробное, хорошо написанное 500-страничное руководство в электронном виде. Бумажный учебник – это здорово, но поиск по ключевым словам гораздо легче выполнять среди бит, чем среди квадратных сантиметров, да и лес стоит поберечь.

За и против

Итак, рядом с бесплатным *OpenOffice.org* – найдется ли место комплекту ценой около $70? Пользователи с опытом работы в *Excel*, особенно любители макросов, найдут этот вариант весьма заманчивым. Его способность уверенно импортировать и экспортировать документы из разных версий комплекта независимо от первоисточника может стать дополнительным бонусом для тех, кто решился перейти с Windows на Linux.

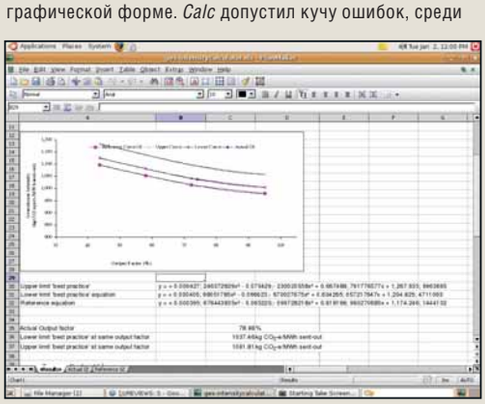
Но если таблицы для вас не единственная и окончательная цель, то градус рекомендаций существенно снижается. Текстовый процессор сравним с *Writer*, вот и прикиньте, стоит ли переплачивать ради ускорения работы (перед покупкой можно попробовать демоверсию). Но отсутствие модуля для презентаций – это серьезно, и мы не можем рекомендовать *SoftMaker 2006* от всей души. Если *OOo* необходим вам из-за приложения *Impress*, то и остальных его частей для жизни хватит.

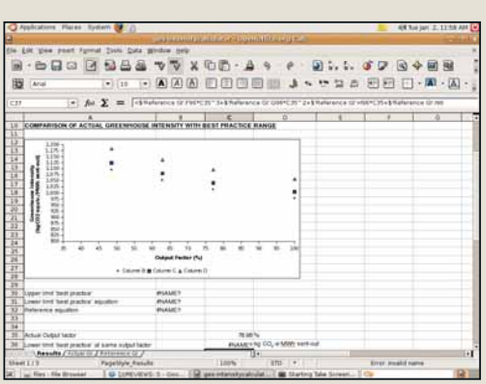
Тем не менее, стабильность, быстрота, безупречная поддержка *Excel* и встроенная база данных делают комплект весьма привлекательным. Если бы встал выбор между покупкой этого дуэта и борьбой с *KOffice*, мы бы уже принялись копить денежки. LXF

PlanMaker против Calc

PlanMaker (слева) и *OpenOffice.org Calc* (справа) попытались смоделировать расчета вывоза алмазов, представленную австралийским правительством. Таблица содержит большое количество данных и макросов VBA для представления результатов вычислений в числовой и графической форме. *Calc* допустил кучу ошибок, среди которых исчезновение самого графика и невозможность проследить за равенствами в разгар работы.

Это наглядно показывает разрыв между двумя приложениями, *PlanMaker* действительно воспроизводит таблицы более уверенно, чем Calc.





LINUX FORMAT Вердикт

SoftMaker Office 2006

Разработчик: SoftMaker Software
Сайт: www.softmaker.com
Цена: $69.95 (или €69.95 с НДС)

Функциональность	7/10
Производительность	9/10
Простота использования	6/10
Документация	9/10

» *На фоне бесплатных приложений выглядит неплохо. Для выживания, необходимо обзавестись модулем презентаций.*

Рейтинг 7/10

VMware Workstation

Впечатлениями о *VMware Workstation 6* готов поделиться **Нейл Ботвик** – у него редко бывает меньше дюжины дистрибутивов одновременно.

Вкратце...

» Флагман виртуализации VMware для настольных машин: новые функции и больше поддерживаемых дистрибутивов. См. также: *Qemu* и *VMware Server*.

Это все Майк Сондерс. Что ни месяц, он заливает как минимум один полный дистрибутив на LXFDVD – извольте пробовать! Охнуть не успеешь, как винчестер переполняется, а загрузочное меню упирается в пол. А с виртуальной машиной можно опробовать дистрибутив без мультизагрузки: в окне запускается компьютер с собственной памятью (и постоянной, и оперативной) и сетевыми подключениями.

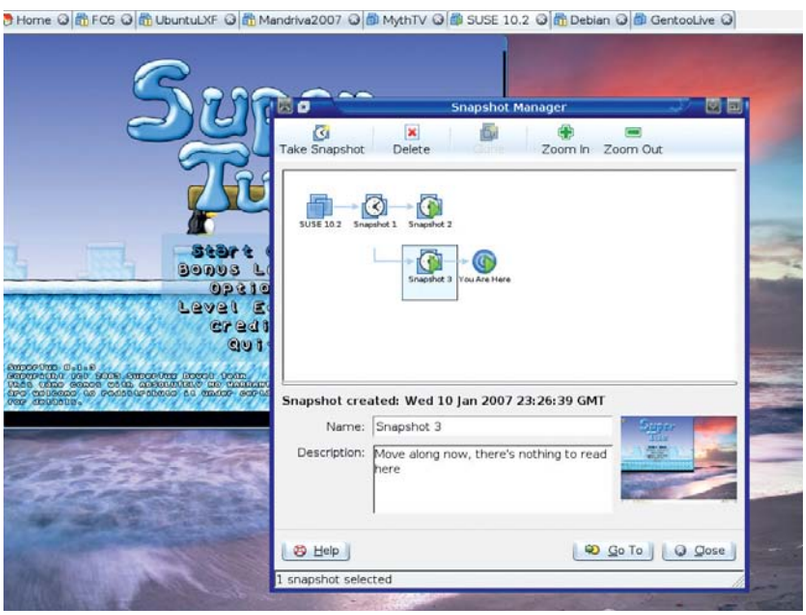
Областей применения виртуальных машин (ВМ) немало: хостинговые компании экономят деньги на физических компьютерах, разработчики безопасно тестируют ПО, образовательные учреждения предоставляют студентам стандартное окружение. Известнейший производитель ВМ – VMware Inc – только что сняла покрывало с *VMware Workstation 6*. Хотя на вид новая версия неотличима от предыдущих, да и работает с ВМ, созданными до нее, изменений по сравнению с *Workstation 5.5* немало. Одно из них состоит в том, что обновление больше не будет односторонним процессом: можете создавать ВМ на *Workstation* 6, затем запускать их на старых версиях.

Теперь виртуальные машины могут трудиться в автоматическом режиме после остановки основной программы *Workstation*. ВМ работает в фоне, с удаленным доступом как с локального хоста, так и с другого компьютера. Это существенное добавление, особенно при разработке ПО без графического интерфейса (сетевое ПО или программы командной строки) на гостевой ОС. Прежний лимит 4 ГБ ОЗУ на все одновременно работающие ВМ полностью снят, а для каждой отдельной – повышен до 8 ГБ.

Теперь и Vista

Кроме того, VMware увеличила количество поддерживаемых систем – например, Windows-версия теперь работает под Vista. Ряды поддерживаемых Linux-дистрибутивов пополнили OpenSUSE 10.2, Mandriva 2007 и Ubuntu 6.10. Этот список имеет значение лишь для получения поддержки от VMware, так как *Workstation* работает на большинстве дистрибутивов – мы пользуемся ей на Gentoo (который не поддерживается) начиная с 3-й версии.

Более существенно включение 64-битной версии. Прежние были исключительно 32-битными и очень ломкими на 64-битных хостах. Имейте в виду, что хотя работа 32-битной гостевой системы на 64-битном хосте возможна, для запуска «64 на 64» необходим как минимум Athlon 64 типа D. Лучше сперва убедиться в том, что *Workstation* 6 работает на вашем оборудовании, на пробной версии, поскольку полная стоит денег.



› **Снимки состояния позволяют вернуть ВМ в любое прошлое.**

Улыбочку, пожалуйста!

Важная для ВМ функция – способность делать снимки состояния системы. В *Workstation* снимок может быть сделан по нажатию кнопки, но можно настроить ее так, чтобы состояние ВМ фиксировалось при каждом выключении, или чтоб снимок загружался при каждом запуске. Это идеально для обучающих систем или машин общественного пользования.

Возможность разрабатывать код в хост-системе и тут же испытывать его в виртуальной – это здорово, но раньше приходилось всякий раз компилировать программу, а затем посылать ее на ВМ. В *Workstation 6* появился модуль Eclipse, с помощью которого код инсталлируется на ВМ прямо из среды разработки. Эту функцию можно сделать доступной по умолчанию во время установки.

О производительности судить рано, так как мы рассматривали бета-версию с включенной отладочной информацией, а это ощутимо снижает скорость; но она была сравнима с 5.5, а VMware утверждает, что производительность по сравнению с прежней версией не изменилась. Гостевые ОС, как правило, работают со скоростью в пределах нескольких процентов от «родной», хотя скорость дискового доступа существенно снижается при использовании в качестве виртуального жесткого диска файла-образа.

Если вам просто захотелось взглянуть на последний дистрибутив Майка, тут хватит и альтернативы вроде *VMware Server* или *Qemu*. Но если виртуальные машины нужны вам в профессиональной деятельности или сфере разработки ПО, возросшая мощь *Workstation* может оказаться неоценимой, а новые функции шестой версии – автоматические ВМ, интеграция с *Eclipse*, поддержка 64-битных хостов – делают обновление весьма привлекательным. Мы бы приобрели ее хотя бы из-за способности работать без графики. LXF

LINUX FORMAT Вердикт

VMware Workstation 6 beta

Разработчик: VMware Inc

Сайт: www.vmware.com

Цена: Требует уточнения (VMware Workstation 5.5 стоит от $189).

Функциональность	9/10
Производительность	7/10
Простота использования	8/10
Цена	7/10

» *Крутые обновления. Заслуживает серьезного внимания – ведь вам нужна мощь?*

Рейтинг 8/10

wxWidgets 2.8

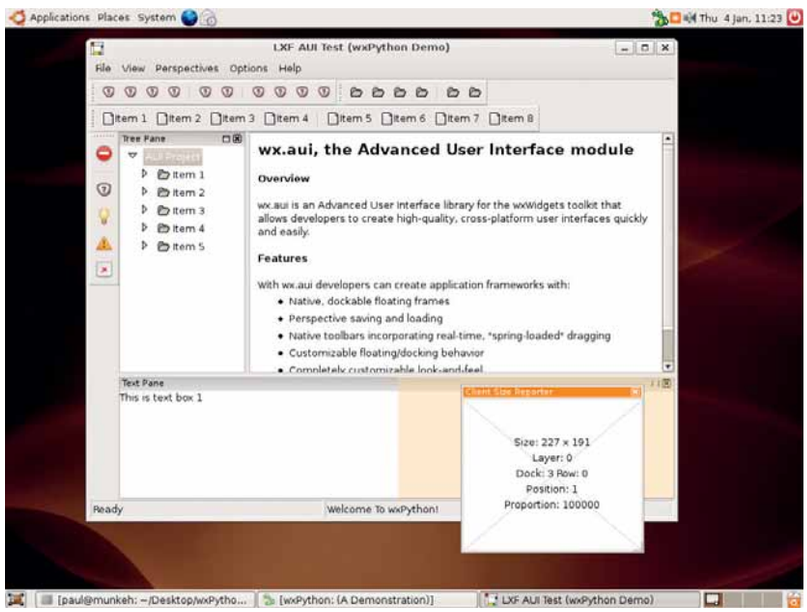
Темная лошадка среди GUI-инструментариев, теперь еще и с визуальным конструктором интерфейсов. **Пол Хадсон** спихивает с экрана Qt с GTK и запускает Python…

Вкратце...

» Межплатформенный GUI-инстументарий на C++, с привязками к Python and Perl. См. также: *Qt* и, конечно, *GTK*.

Час языка программирования Ruby пробил лишь с появлением среды разработки Rails. В случае с *wxWidgets* все наоборот: эта библиотека обрела популярность лишь после создания *wxPython*. Пока KDE/*Qt* и Gnome/GTK пожинали лавры, *wxWidgets* потихоньку росли над собой, внедряя улучшение за улучшением.

Как и *Qt*, *wxWidgets* – межплатформенный инструментарий для создания графических пользовательских интерфейсов, написанный на C++. Опять же, как и *Qt*, *wxWidgets* имеет массу дополнительных функций вроде поддержки потоков, доступа к базам данных и т.п. Но, в отличие от *Qt*, ядро *wxWidgets* – оно отвечает за рендеринг всего GUI – целиком полагается на ОС, воспроизводящую интерфейс, а значит, приложения *wxWidgets* будут почти неотличимы от «родных» [*Qt* и *GTK+* эмулируют внешний вид и поведение интерфейсных элементов целевой платформы, делая это более или менее удачно, – прим. ред.]. Еще важнее то, что у *wxWidgets* более либеральная лицензия, чем у *Qt* – модифицированная LGPL, позволяющая создавать с помощью инструментария приложения с закрытым исходным кодом, тогда как версия GPL для *Qt* требует непременного раскрытия кодов.



› **В *wxAUI* передвигаемый фрейм становится полупрозрачным, предварительно указывая место предстоящего размещения.**

«Максимум функций при минимальных усилиях программиста.»

Все это немаловажно, но для нас главное – энтузиазм, с которым сообщество *wxWidgets* поддерживает программу, и с версией 2.8 этот энтузиазм не ослабел. Среди новинок – полностью переработанный элемент rich text с API в стиле стека, напоминающим HTML; инструмент выбора цвета (color picker), выводящий выбранный цвет на кнопку; и элемент «гипертекст», открывающий браузер пользователя при щелчке по гиперссылке. Такое мог бы запрограммировать для себя каждый, а *wxWidgets* взял да и ликвидировал лишнюю мороку.

Эти функции дополняются десятками других мощных инструментов, например, неизменно популярным менеджером «мышиных» жестов, что позволяет *wxWidgets* прочно удерживать второе место по простоте использования среди GUI API после *Gambas*. Все новшества 2.8 работают надежно, но гвоздь программы – несомненно, wxAUI, что означает WX Advanced User Interface (Продвинутый интерфейс). Он призван устранить неразбериху, созданную *wxIFM*, *wxDockIt* и *wxFrameLayout*, пытающимися сделать одно и то же. Цель wxAUI – предоставить полную поддержку передвижения и размещения фреймов и панелей инструментов в пределах интерфейсов *wxWidgets*, а кроме того – упростить сохранение и загрузку «перспектив» (сохраненных раскладок).

Новаторский wxAUI

Разработка сложных GUI с использованием фреймов традиционно была нелегкой задачей, поэтому wxAUI можно справедливо считать открытием. Но это не революция: *Qt* представлял (хоть и не такой крутой) QDockWidget еще в *Qt 4.0*, поэтому первенство не за *wxWidgets*. К тому же панелям инструментов wxAUI пока что нужна доводка: например, предстоит научить их рисовать «стрелочки», когда вся панель не умещается в окне.

Еще одна область, в которой *WxWidgets* еще только пробует силы – поддержка письма справа налево. Поддержку арабского в *wxWidgets* для Windows намеревались добавить в рамках проекта Google Summer Of Code, но попытка провалилась. Разработчики *wxWidgets* подобрали обломки, и в новой версии впервые появилась поддержка арабского письма для пользователей *GTK*.

Несмотря на сложности интернационализации, *wxWidgets* продолжает славную традицию: предоставлять максимум функциональности при минимальных усилиях программиста. Пользуетесь ли вы оригинальным вариантом C++ или привязкой к одному из других языков высокого уровня – вы согласитесь, что эта версия определенно стоила ожидания, и несомненно обогатит ваши GUI новыми функциями. Избавиться бы только от поганого чувства, что она смахивает на *MFC* от Microsoft… LXF

Вердикт

WXWidgets 2.8

Разработчик: WXTeam
Сайт: www.wxwidgets.org
Цена: Бесплатно под модифицированной LGPL

Функциональность	7/10
Производительность	9/10
Простота использования	10/10
Документация	10/10

» *С wxWidgets крутым программистом может стать каждый.*

Рейтинг 9/10

ASPLinux 11.2

Готов ли отечественный Linux к развертыванию на отечественных ПК? **Роберт Басыров** пытается взглянуть на последнюю версию ASPLinux глазами простого пользователя.

Вкратце...

» Отечественный настольный дистрибутив на базе Fedora Core. См. также: ALT Linux и непосредственно Fedora.

«Все начиналось с черной кошки». Красивая фраза, но суть отражает не полностью. Конечно, дистрибутив Black Cat Linux послужил основой для создания линейки *ASPLinux*, но нельзя говорить, что этот популярный в России дистрибутив начался с приходом в компанию *ASPLinux* Леонида Кантера и Александра Каневского. До этого объединения был выпущен *ASPLinux* Release Candidate 3, разрабатывался собственный менеджер загрузки *ASPLoader*. Тем не менее, первые массовые дистрибутивы *ASPLinux* «подхватили» нумерацию версий Black Cat Linux, а это говорит о значении упомянутых программистов в совместном продукте.

Начиналось все относительно недавно

На тот период, когда возникла идея создания *ASPLinux* (а это 2000 год), самым качественным с точки зрения сборки пакетов и удобства в работе считался Red Hat 6.2. Однако, его инсталлятор требовал от пользователя определенных знаний и опыта. Примерно в то же время вышел дистрибутив Corel Linux, который был основан на ограниченном наборе пакетов Debian (только рабочая станция), но имел очень красивый и простой в использовании установщик, доступный даже для начинающих пользователей. Поэтому внутри компании SWSoft («прародителя» *ASPLinux*) возникла идея создания дистрибутива, который сочетал бы в себе функциональность и качество сборки Red Hat с простотой и удобством Corel Linux. Исходные тексты установщика Corel были доступны по лицензии GPL, поэтому он был частично использован при создании инсталлятора *ASPLinux*. Сначала выпускались только универсальные дистрибутивы (которые можно было использовать и на рабочем столе, и на сервере) но, начиная с версии 7.3, в *ASPLinux* была выделена серверная ветвь, так и названная – *ASPLinux* Server.

Этой же линии (углубление различий между настольной и серверной версиями) компания придерживается сейчас и намерена придерживаться в будущем. В настольных версиях будет осуществляться поддержка самого современного оборудования и использоваться последние версии рабочих сред, а в серверную будут включаться стабильные и проверенные компоненты, обеспечиваться расширенная

› **Вездесущий *Clearlook* добрался и до *ASPLinux*.**

техническая поддержка. При этом основным параметром оценки своих продуктов компания считает стабильность. Все новомодные «навороты» хороши, но если из-за них потеряется хотя бы часть пользовательских данных, вряд ли это кого-то обрадует.

О популярности дистрибутива можно частично судить по форуму сообщества *ASPLinux*, в котором с нетерпением обсуждались сроки выхода новой версии – 11.2. Что касается сравнений с другими дистрибутивами, то сложно сказать, какая из разработок лидирует в России, так как не существует общепринятой и открытой статистики. (Да и возможна ли она для свободного ПО, где с одного диска можно установить систему на десятки и сотни компьютеров?) Единственное, что можно сказать точно – то, что *ASPLinux* явно лидирует по коробочным продажам через сеть компании 1С.

Обнаружено новое устройство?

Поддержка нового оборудования – одна из постоянных проблем для разработчиков любых дистрибутивов. Если драйвера для Windows создатели видео- и звуковых карт и прочего компьютерного «железа» разрабатывают сами, то в Linux, как известно, все обстоит по-другому. Постоянное сотрудничество инженеров *ASPLinux* с ведущими производителями компьютеров позволяет новой версии дистрибутива определять и корректно настраивать при инсталляции самое популярное на сегодняшний день оборудование: контроллеры SATA, видеокарты ATI и NVIDIA (причем с фирменными драйверами), звуковые карты.

Нельзя сказать, что у пользователей дистрибутивов *ASPLinux 11.2* не бывает проблем по аппаратной части. Были они и у нас, но все это не выходило за рамки типовых проблем Linux. Что поделаешь, если Canon не пишет драйвера для своих принтеров под Linux?

Кросс-платформенные приложения

Несмотря на заметное противостояние закрытого и открытого ПО, все большее развитие получают кроссплатформенные приложения. Для обеспечения работы таких приложений в новом дистрибутиве предусмотрена интеграция с Mono, открытой реализацией Microsoft .NET. В состав *ASPLinux 11.2* входит несколько приложений, написанных с использованием Mono:

» персональная поисковая система Beagle;
» фотоальбом F-Spot;
» менеджер заметок Tomboy.

Персональная поисковая система *Beagle* интегрирована в графическую среду Gnome и позволяет найти введенную фразу в документах *OpenOffice.org*, сообщениях электронной почты (базы Evolution), диалогах IM-клиента *Gaim*, кэше браузера *Firefox*.

ASPLinux, как и его родитель, Fedora Core, тяготеет к Gnome. Любителям KDE приходится устанавливать свои «красивости» вручную. У нас не получилось с ходу инсталлировать эту рабочую среду даже с использованием штатных средств: менеджер пакетов не нашел библиотеку *libpostproc*. Пришлось искать, скачивать и ставить пакет вручную. Дело хозяйское, но с таким подходом о массовости использования *ASPLinux* говорить не приходится.

Есть еще один не совсем понятный момент в работе с пакетами – штатная утилита *Yum* не позволяет автоматически использовать установочный DVD для поиска и инсталляции пакетов. Выбор профиля Local не обеспечивает обращения к диску, для этого необходимо редактировать конфигурационные файлы вручную.

Однако надо заметить один положительный момент. *Yum Extender*, при использовании его с терминала, поддерживает докачку. После обрыва связи *Yum* сам подхватывает загрузку недогруженных файлов. Для «счастливых» обладателей модемов это бесценное свойство. Жаль, что при использовании графического интерфейса такого не происходит.

3D desktop

В наше время невозможно обойти в обзоре дистрибутива тему трехмерного рабочего стола, точно так же, как трудно представить себе современный дистрибутив без этой функции. В *ASPLinux 11.2* включен *Xgl* от компании Novell, как самая стабильная, по мнению разработчиков, реализация [Fedora Core использует *AIGLX*, разработанный Red Hat, – прим. ред.]. Необходимые для работы *Xgl* проприетарные драйвера входят в состав дистрибутива, так что у вас не должно возникнуть проблем, если конечно, ваше оборудование достаточно современное.

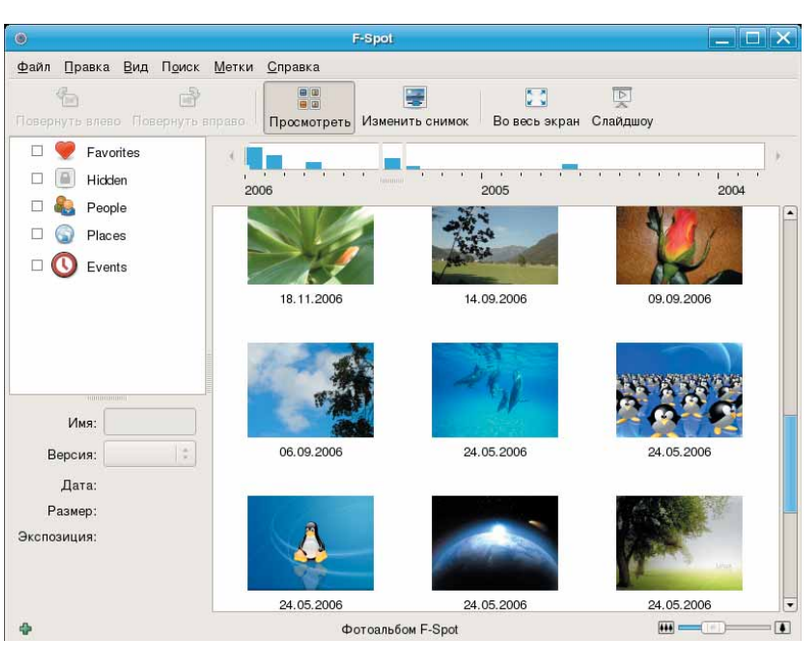
Появление трехмерного рабочего стола – не единственное обновление дизайна в *ASPLinux 11.2*. Основной темой оформления теперь является *Clearlook*. В сочетании с «новым лицом» экрана входа в систему и новыми обоями рабочего стола интерфейс стал более приятным для глаз.

Но и трехмерность рабочего стола, и смена основной темы – это внешние, видимые изменения. Для пользователя гораздо важнее будет другое – повышение производительности. Несколько важных составляющих рабочего окружения GNOME, как то: отрисовка текста, выделение памяти, некоторые отдельные приложения, теперь работают значительно быстрее. В результате уменьшается время загрузки рабочей среды, программы в целом быстрее реагируют на ваши действия, а это значительно повышает комфортность работы.

Для владельцев ноутбуков

В *ASPLinux 11.2* была добавлена пара опций, которые облегчат жизнь владельцам ноутбуков. Прежде всего, это *NetworkManager*, который избавляет пользователей от необходимости вручную переключаться на новую сеть при перемещении из одного помещения в другое. *NetworkManager* может автоматически переключится с проводной сети на беспроводную и обратно.

Функция управления питанием для ноутбуков – одна из самых важных. Новый пакет gnome-power-manager позволяет настраивать реакцию операционной системы на такие события, как критический разряд батареи, закрытие крышки и другие. Пользователь может настроить для каждого из этих событий свое действие: переход как в спящий режим (suspend), так и ждущий (hibernate).



› Менеджер фотографий F-Spot – хороший пример использования Mono.

IP-телефония и видеоконференции

Голосовое общение через Интернет приобретает все большую популярность. Приложение *Ekiga*, входящее в *ASPLinux 11.2* и ранее известное как *GnomeMeeting*, теперь поддерживает не только протокол H.323 (Microsoft NetMeeting и другие приложения на платформе Windows), но и SIP, который используется в Google Talk и программной *ATC Asterisk*. Ekiga также может поддерживать сервис STUN, решающий проблемы с NAT-маршрутизаторами. Если говорить простым языком, то с помощью *Ekiga* можно звонить и принимать звонки, не имея публичного IP-адреса и не настраивая переадресацию портов на шлюзе.

И напоследок

Есть еще несколько мелких улучшений, которые, возможно, и не так значительны, но интересны. В *Evolution* была реализована поддержка общих календарей с помощью протокола CalDAV. Этот протокол поддерживается различными серверами совместной работы, включая свободно доступный сервер Hula. Кроме этого, *ASPLinux 11.2* может создавать и использовать логические тома *LVM*. Установка операционной системы на массив логических томов *LVM* позволяет гибко управлять дисковым пространством и при необходимости расширять объем дискового пространства «на лету».

Резюме

ASPLinux – удобный и надежный дистрибутив. Те, кто уже работал с ним, вряд ли сменят свой ASP на что-то другое. Для новичка этот дистрибутив станет точно таким же полигоном, как и любой другой. Но надо отметить один момент: *ASPLinux* – отечественная разработка. Он глубже других дистрибутивов русифицирован и документирован, а это далеко не малый плюс для тех, кто только вступает в мир Linux. LXF

LINUX FORMAT Вердикт

ASPLinux 11.2

Разработчик: ASPLinux
Сайт: www.asplinux.ru
Цена: 775 р. (версия Standard)

Функциональность	8/10
Производительность	8/10
Простота использования	7/10
Документация	8/10

» Неплохой выбор для русскоязычных пользователей, особенно не имеющих опыта работы в Linux.

Рейтинг 8/10

Kerio Mail Server 6.3

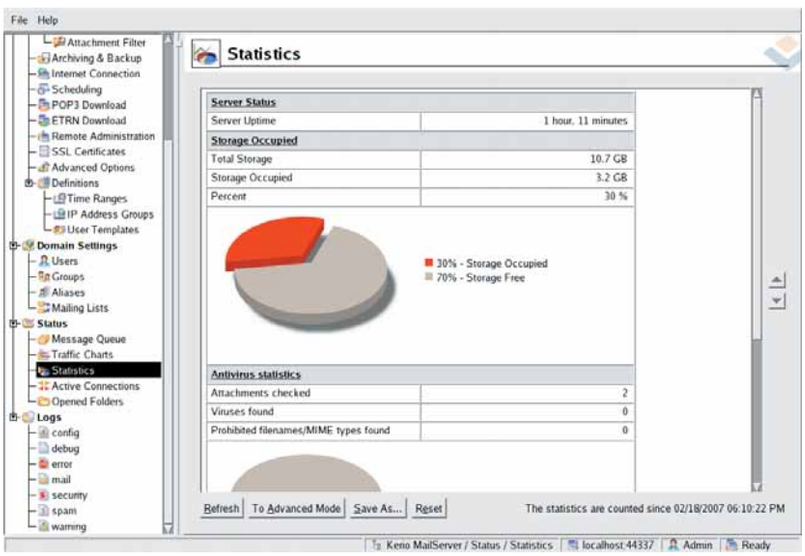
Вдоволь наигравшийся с макросами m4 **Андрей Тусеев** рассматривает альтернативу Sendmail, которую можно настроить одной мышью.

Вкратце...

» Коммерческий почтовый сервер для Linux с функцией web-доступа. См. также: *Axigen*.

Пользователи Linux вряд ли могут жаловаться на недостаток почтовых серверов. Для идейных противников старого доброго *Sendmail* есть *Postfix*, qmail и множество других альтернатив; обеспечение доступа к почтовому ящику можно поручить как крохам вроде *popa3d*, так и тяжеловесам: *Cyrus* и прочим. Для желающих иметь доступ к письмам даже из интернет-кафе, затерянного где-то в предгорьях Анд, есть *SquirrelMail*. Богатый ассортимент – это прекрасно, но есть и обратная сторона медали – администратору, отвечающему за почтовый шлюз, необходимо изучить (а в дальнейшем – поддерживать) несколько различных продуктов, каждый из которых обладает собственными правилами (и тонкостями) настройки.

Решения вроде *Kerio Mail Server* (KMS) стремятся устранить этот недостаток. Пришедшие из мира Windows, они являются комбайном типа «все в одном» и стремятся предоставить понятный и единообразный графический интерфейс, упрощающий сопровождение. Мы решили оценить плюсы и минусы этого подхода, установив *Kerio Mail Server 6.3.1* на систему под управлением CentOS 4.4, двоично совместимую с RHEL.



› **Круговые диаграммы порадуют «пиджаков», но ничего не скажут «технарям».**

Первое знакомство

Продукт распространяется в виде RPM-пакета и поддерживает практически все распространенные серверные системы (можно отметить разве что отсутствие модного нынче Ubuntu 6.06 LTS): Red Hat Enterprise Linux 3/4, Fedora Core 4/5, SUSE Linux 10.0/10.1. Огорчает только отсутствие документации – в дистрибутиве ее нет совсем, а информация на сайте посвящена, в первую очередь, версии для Windows (впрочем, принципиальных отличий не так уж и много). После установки пакета вам будет предложено прочитать краткую справку, остановить сервисы Sendmail/Postfix и запустить утилиту *cfgwizrd* для первичной настройки сервера. *Cfgwizard* использует текстовый оконный интерфейс, что слегка удивляет – от коммерческого продукта обычно ждешь приятных графических мастеров. Тем не менее, со своей задачей мастер *cfgwizard* справляется без проблем. Вас попросят указать имя первичного почтового домена и его почтового хоста, которое, естественно, должно указывать на машину с запущенным *Kerio Mail Server*. Настройка DNS-зон в обязанности мастера не входит, так что вам следует позаботиться об этом заранее. На следующем шаге создается учетная запись администратора почтового сервера, затем – указывается каталог для хранения почтовых сообщений, и – вуаля! Ваш сервер практически готов к работе – осталось только запустить его командой **/etc/init.d/keriomailserver start**. В целом, процесс первичной настройки сервера в Linux повторяет таковой в Windows, что весьма порадует мигрантов с этой ОС.

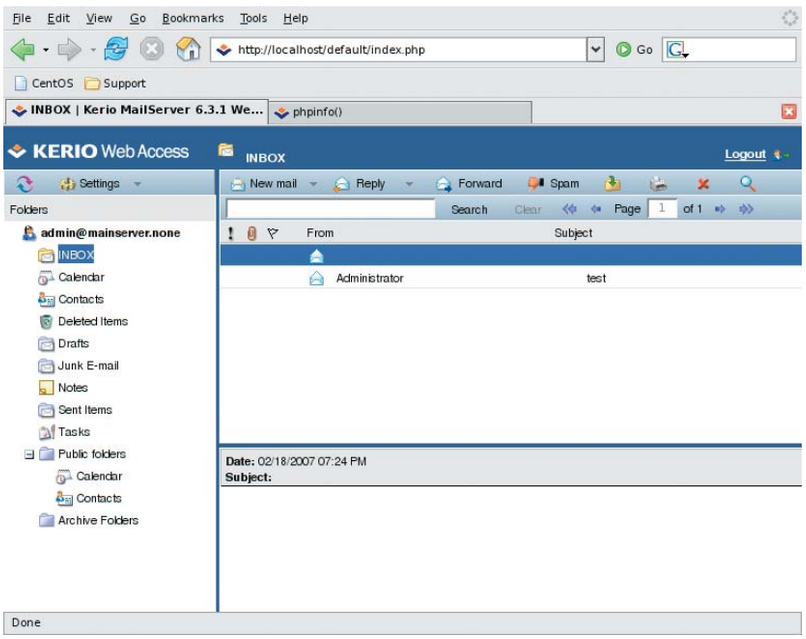
Гюльчатай, открой личико!

Управление работающим сервером осуществляется посредством графической консоли, запускаемой командой kerioadmin. Если разработчики *Kerio Mail Server* задавались целью выдержать ее в духе традиционного для Unix графического инструментария (Motif, конечно... А вы что подумали?), то им это вполне удалось. Несмотря на то, что kerioadmin использует *Qt*, элементы интерфейса выглядят архаичными и плохо согласуются с современным рабочим столом. Продукт поддерживает 12 языков, включая русский, но опцию для их смены найти не так-то просто, особенно человеку, не владеющему английским (впрочем, среди администраторов их становится все меньше и меньше).

Для входа в систему необходимо использовать имя пользователя и пароль, указанные на этапе первичной настройки. После этого откроется окно, представляющее все возможности сервера в виде древовидного списка. Итак, что же предлагает нам Kerio?

Во-первых, стандартный набор служб: SMTP, POP3, IMAP, LDAP, HTTP со всеми безопасными разновидностями. «Самоподписанный» (self-signed) сертификат можно сгенерировать при помощи мастера. Во-вторых, виртуальные домены (способ по-разному трактовать учетные записи в зависимости от того, к какому домену они принадлежат) с поддержкой служб каталогов Active Directory, Apple Open Directory и авторизацией через PAM/Kerberos, так что в этой области *Kerio Mail Server* вряд ли уступает своим закрытым конкурентам. Более того, KMS позволяет настроить разные методы авторизации для различных доменов: например, один домен авторизуется в одной службе каталогов, другой домен авторизуется непосредственно через PAM, и т.д. Помимо этого, предлагается контент-фильтр, включающий антивирус (интегрированный – от *McAfee* или внешний – *AVG*, *NOD32*, *Sophos*, *Symantec* и, конечно же, *ClamAV*. Выбор за вами, только имейте в виду, что в случае использования внешнего антивируса об обновлении баз придется заботиться самостоятельно), контроль вложений (в списке потенциально опасных расширений по

умолчанию почему-то не активировано *.EXE) и антиспам. О последнем компоненте скажем чуть более подробно. Встроенный спам-фильтр KMS поддерживает систему рейтингов в стиле SpamAssassin, «черные» списки (как RBL, так и пользовательские; имеется возможность добавлять баллы в рейтинг вместо отказа в получении письма), «белый» список (по IP), технологии Caller ID и SPF, а также несколько примитивную систему пользовательских правил. Серые списки в явном виде отсутствуют – их заменяет механизм подавления спамерской активности, внедряющий дополнительные циклы ожидания в общении MTU/MTA. Судя по всему, спам-фильтр построен на базе SpamAssassin, что, на наш взгляд, вполне соответствует философии продуктов класса *Kerio Mail Server*: «предоставлять стандартные решения, но в удобной форме». Тонкая настройка параметров фильтрации не предусмотрена, но с помощью имеющихся средств можно вполне надежно защититься от нежелательной корреспонденции.



› ***Kerio Web Access*** **– подозрительно напоминает OWA, но поддерживает Mozilla.**

Kerio Mail Server может не только выступать в роли MTA, принимая всю почту, адресованную вашему домену, но и просто периодически загружать сообщения из другого почтового ящика, поддерживаемого, например, вашим провайдером. Скачанные письма можно сортировать на основе заголовков. Поддерживаются протоколы POP3 и ETRN – последний используется не так уж часто, но лишняя функция, очевидно, не помешает.

Отдельного упоминания заслуживают еще две функции – архивация писем (Archiving) и резервирование (Backup). KMS позволяет хранить всю корреспонденцию в структурированных архивах, так что всегда можно найти и «поднять» старое/забытое/удаленное письмо. Это будет полезно в организациях, в которых учет переписки имеет принципиальное значение (а где не имеет?).

Резервирование также реализовано просто. По сути, достаточно указать каталог, в котором следует сохранять резервные копии (конечно, крайне желательно, чтобы он физически находился на отдельном компьютере или хотя бы диске), а набор предустановленных правил резервирования позаботится обо всем остальном. Поддерживаются как полные, так и дифференциальные резервные копии.

Пожалуй, единственная область *Kerio Mail Server*, выпадающая из сформулированного выше принципа «как у всех, но удобнее» – это ведение статистики и файлов журналов. И то, и другое откровенно слабо и практически не поддается настройке. Вероятно, сводная картина происходящего на сервере понравилась бы Большому Боссу, но она не несет достаточной информации для администратора, желающего произвести сколько-нибудь серьезный анализ. Журналы событий представлены в большом количестве, но содержащаяся в них информация сама по себе не может быть использована для оценки происходящего. Кнопка To Advanced Mode (*см. рис.*), казалось бы, должна помочь, но не тут-то было: она всего лишь дополняет отчет еще более бесполезным данными, такими, как суммарный размер почтовых сообщений, прошедших через сервер. Не хватает настроек детализации журналов, потому как на одну сессию с сервером в журнале отведена только одна строка со множеством параметров, по которой очень сложно отследить, что именно пытался сделать клиент.

Выходим в Сеть

Конечно, универсал, на роль которого претендует *Kerio Mail Server*, не может обходиться без web-почты. Клиент Kerio Web Access являет собой слабую попытку сымитировать Outlook Web Access, предоставляемый Microsoft Exchange, но его внешний вид гораздо более приятен, чем у *SquirrelMail* в настройке по умолчанию. KWA, так же как и OWA, не любит Opera, но к счастью, поддерживает Mozilla. Порадовало также наличие календаря, как группового, так и персонального.

Для поддержки web-почты не нужно конфигурировать Apache: web-сервер встроен в продукт, и его можно настроить на использование любого порта или группы портов. *Kerio Web Access* написан на PHP (KMS содержит свой собственный интерпретатор версии 4.3.11). Исходный код, за исключением модулей расширения, доступен в человеко-читаемой форме, что, вероятно, позволит (при необходимости) доработать KWA под нужды организации.

Заключение

Следует признать, что функциональность данного продукта (при сравнительно небольшой цене) впечатляет. Конечно, в нем нет ничего такого, что не смог бы настроить знающий администратор на базе стандартных средств, но зато и не надо тратить время на их изучение. *Kerio Mail Server* ориентирован на компании с гетерогенными сетями/серверами и средним (или даже небольшим) IT-бюджетом. Он также должен понравиться мигрантам с Windows, особенно тем, кто уже использовал продукты *Kerio* на этой платформе. LXF

LINUX FORMAT Вердикт

Kerio Mail Server 6.3

Разработчик: Kerio Technologies Inc.
Сайт: www.kerio.com/ru
Цена: от €400 (лицензия на 10 пользователей)

Функциональность	6/10
Производительность	6/10
Простота использования	8/10
Цена	6/10

» *Должен понравиться новичкам и мигрантом, но вы вряд ли захотите заменить им уже настроенный Sendmail или Postfix.*

Рейтинг 6/10

Сравнение

Каждый месяц мы анализируем для вас тысячи программ — а вы можете отдохнуть!

LINUX FORMAT на DVD

Клиенты BitTorrent

Во избежание узких мест при скачивании нового дистрибутива, **Морис Келли** интересуется клиентами *BitTorrent* последнего урожая.

Paul Blachford

Про наш тест...

Хороший клиент BitTorrent должен предоставлять пользователю возможность одновременно поддерживать множественные загрузки, так что мы тестировали функции контроля за множеством загрузок и назначения приоритетов конкретным торрент-файлам. Важна также способность приостанавливать и перезапускать закачки, или даже возобновлять закачку, первоначально начатую другим клиентом. Дополнительно мы проверяли клиенты на способность создавать торрент-файлы, чтобы пользователи могли рассылать собственные большие файлы.

Тестовая среда для этого Сравнения включала несколько систем. Первая – компьютер с Ubuntu 6.10 (Edgy), запущенным на Pentium M 750 с 1 ГБ ОЗУ, мы использовали ее главным образом для тестирования графических клиентов. Вторая – система Debian Sarge, работающая на Celeron 2.66 ГГц с 512 МБ ОЗУ, используемая для тестирования клиентов, работающих из командной строки или через web-интерфейс.

Наш выбор

Azureus с. 20
BitTornado с. 21
BitTorrent с. 19
Deluge с. 19
KTorrent с. 21
TorrentFlux с. 22
Transmission с. 22

Нравится оно вам или нет, но распространение файлов с помощью пиринговых (peer-to-peer) сетей стало неотъемлемой частью современного компьютерного мира. Со времен вспышки Napster в 2000 г. число протоколов передачи файлов росло как на дрожжах. Организации, подобные RIAA (Американская ассоциация индустрии звукозаписи) и MPAA (Американская кинематографическая ассоциация), делали все возможное, чтобы остановить прогресс; в итоге, выжили только сильнейшие протоколы. Один из них – *BitTorrent*, разработанный в 2001 г. Брэмом Коэном [Bram Cohen]: он оказался особенно полезен для распространения больших файлов.

Первоначальный «основной» клиент *BitTorrent*, ориентированный на командную строку, был написан на Python и делал свою работу очень хорошо. Он доступен и в наши дни в виде обновленной инкарнации, но соседствует со множеством других клиентов. В этом Сравнении мы взглянем на «молодую поросль» приложений, чтобы посмотреть, как они поживают не просто как клиенты, но как полноценные менеджеры закачек *BitTorrent*.

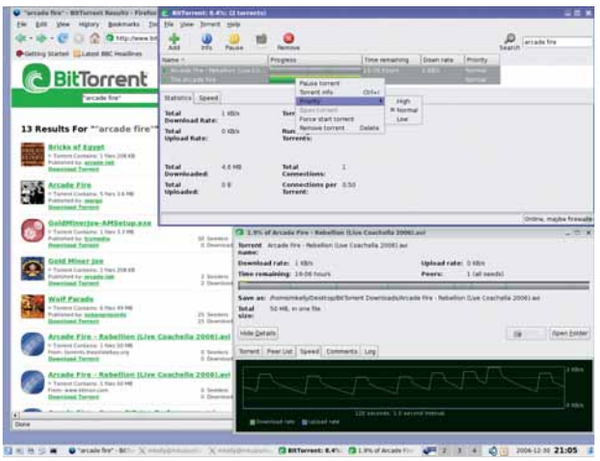
И как это работает?

Чтобы распространять чтыо-то через *BitTorrent*, создается торрент-файл, содержащий информацию о «трекере» и о файлах, включенных в этот «торрент». Торрент-файл затем публикуется и загружается конечными пользователями. Когда кто-нибудь открывает торрент-файл в клиенте, то устанавливает соединение с трекером, то есть приложением, которое предоставляет файл наряду с информацией о других клиентах, у которых можно получить содержимое «торрента». Первый клиент – это «сидер» (seeder), и все последующие клиенты будут взаимодействовать с этим хостом для получения частей «торрента». Когда первоначальная раздача (seeding) завершится, ответственность за распространение разделяется среди тех, кто уже скачал части «торрента». Благодаря таким свойствам *BitTorrent* был принят рядом Linux-дистрибутивов, желающих разгрузить свои FTP-серверы.

За прошедшее с момента создания время данная технология не стояла на месте, и в наши дни предоставляются такие функции, как бестрекерные «торренты», позволяющие еще больше децентрализовать распространение. Клиенты, которые используют распределенную хэш-таблицу (DHT, distributed hash table), способны взаимодействовать непосредственно друг с другом для получения информации о файле, а не полагаться на единую точку отказа.

BitTorrent

«Официальный» не значит «лучший»…



› Панель поиска отправляет вас на сайт *BitTorrent*, но хорошо ли это – вопрос спорный.

Довольно велики шансы, что устанавливая *BitTorrent*-пакет(ы) из вашего дистрибутива, вы получите официальный клиент *BitTorrent* – также известный как «основной» (mainline) клиент. Это продукт развития первоначального клиентского ПО, написанного Брэмом Коэном; на данном этапе – версия 5.0.3. Выдержанный в духе философии Unix, пакет содержит ряд отдельных инструментов: прежде всего это *BitTorrent* для закачки и *Maketorrent* для создания торрент-файлов.

Есть консольные средства и для загрузок, и для создания торрент-файлов; выделенные инструменты *Launchmany* позволяют консольным инструментам оперировать множеством «торрентов» за раз. Включено также трекерное ПО.

Графический интерфейс модернизирован и оперирует множеством «торрентов» в одном окне, а также поддерживает панель поиска, предоставляющую доступ к поисковой машине на сайте www.bittorrent.com – увы, его содержимое не впечатляет, да и машина слабовата. «Торрентам» можно назначать приоритеты: просмотрев файлы, установить флаг «Download First» тем, которые вы хотите получить раньше других. Если нижняя панель открыта, в ней отображается аккуратная статистика и график скорости для всего приложения, а отдельные «торренты» имеют подобный график в собственных данных.

Конкурентное преимущество официального клиента – в том, что его разработали изобретатели протокола, и он должен бы быть лучшим и наиболее эффективным загрузчиком. Но мы не выявили этого в данном Сравнении. Прямо не верится, что разработчики основного клиента упустили случай сделать его «универсамом» *BitTorrent*. ПО явно относится к «середнякам», web-сайт и поисковая машина – тоже. Он может быть полезен для изолированных загрузок, но для людей, желающих получить полноценный менеджер закачек, эта программа не подходит.

> «Сделан изобретателями протокола, и поэтому должен бы быть лучшим.»

LINUX FORMAT Вердикт

BitTorrent

Версия: 5.0.3

Сайт: www.bittorrent.com

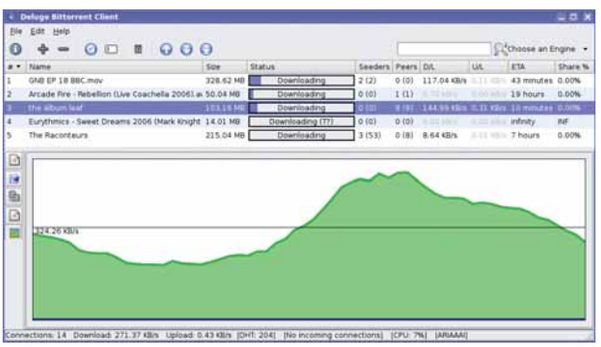
Цена: бесплатно по лицензии BitTorrent Open Source License

» *Довольно средний клиент, который растратил потенциал, заложенный в роли официального клиента.*

Рейтинг 5/10

Deluge

Новичок в классе.



› При семичасовом ожидании, график использования ширины канала слегка скрашивает жизнь.

Намерение *Deluge* [Потоп] – принести клиент *BitTorrent* на основе *GTK* на рабочие столы, например, Gnome и *Xfce*. У него обычные проблемы молодого проекта: недостаток функций и недостаток документации. Да и для освоения это не самый простой клиент. Мы тестировали версию 0.4; должны отметить, что в грядущей 0.5 обещают переписать ряд основных компонентов.

Основное окно предоставляет панель инструментов, поле, содержащее список активных «торрентов», и еще одно поле, с дополнительной информацией по выделенному в данный момент «торренту». *Deluge* не позволяет устанавливать ограничения на скорость загрузки на каждый «торрент», но есть три кнопки на панели инструментов для перемещения «торрента» выше, ниже и на самое дно в системе базовых приоритетов. Для загрузки можно отметить определенные файлы, но назначение приоритетов отдельным файлам не предусмотрено.

Deluge – очередной клиент с системой расширений, предоставляющих дополнительную фунциональность. Среди расширений – монитор загрузки процессора, отслеживание сетевой активности и, для борьбы с силами зла, расширение Anti-RIAA AI. Разработчикам, однако, следует помнить, что на расширения нельзя переваливать основную функциональность: мы заметили, что один из ключевых разработчиков намекает на форуме *Deluge*, что поддержка UPnP – неплохой кандидат в расширения. А вот мы предпочли бы видеть ее в основном пакете.

Даже подпертый расширениями, *Deluge* выглядит скудным. Это довольно милый клиент – он хорошо смотрится и хорошо интегрируется в окружение рабочего стола. Но от клиента, основанного на сторонней библиотеке для поддержки протокола, мы ожидали лучшей функциональности по управлению «торрентами». К сожалению, ему недостает первоклассных функций, к тому же нет документации. Он удобен для разовых закачек, но отсутствие поддержки UPnP и слабая поддержка приоритетов означают, что этот клиент не пригоден для продвинутых пользователей.

LINUX FORMAT Вердикт

Deluge

Версия: 0.4

Сайт: www.deluge-torrent.org

Цена: бесплатно по лицензии GPL

» *Проект не без потенциала, но пока еще слишком юн, чтобы тягаться с большими парнями.*

Рейтинг 5/10

Azureus

Нашпигованный функциями клиент на базе Java.

Впервые появившись на SourceForge в июне 2003 г., проект *Azureus* недавно достиг «главной» версии 2.5.0.0. Благодаря языку Java, *Azureus* – подлинно кросс-платформенное приложение, но, к сожалению, для запуска нуждается в установке виртуальной машины Java (JVM, см. врезку «Клиенты на базе Java»).

При запуске приложения нас встретил «мастер»: он провел нас через некоторые основные настройки и предоставил ряд ссылок на сайты, полезные для начинающих в *BitTorrent* – вроде мелочь, но дружелюбное отношение к пользователю – козырь приложения. По завершении открылось основное окно, состоящее из панели инструментов, панели загружаемых «торрентов» и панели для составленных или раздаваемых «торрентов».

Как это выглядит

По щелчку на кнопке «Open Torrent» откроется подробный диалог, позволяющий добавить торрент-файлы из списка источников, включая файлы и URL. После выбора «торрента» отобразится список содержащихся в нем файлов, и вы сможете отметить для закачки только некоторые из них. После добавления и настройки «торрента» *Azureus* установит соединение и начнет закачку. Если ваш маршрутизатор поддерживает UPnP (universal plug and play), *Azureus* любезно настроит его, чтобы позволить вам принимать входящие соединения. В процессе загрузки проницательному пользователю предоставляется изобилие информации, доступной по двойному щелчку на строке «торрента» в панели загрузки.

При этом появляется новая вкладка с подстраницами, дающими общую информацию и сведения о подключенных «коллегах» (peer) в виде красивого графического представления их роя [swarm – набор компьютеров, участвующих в обмене данными, – прим. пер.], и подгружаются данные по отдельным фрагментам и файлам, составляющим ваш «торрент». Возможно, при скачивании всего-навсего ISO-образа дистрибутива Linux оно и лишнее, зато очень удобно при загрузке большой коллекции файлов.

На вкладке «Info» можно увидеть состояние загрузки отдельных файлов или привести в порядок их приоритеты, чтобы самый интересный трек приплыл первым. Эта страница показалась нам единственным местом, где можно восстановить файл, первоначально отмеченный как игнорируемый в «торренте». Конечно, это несложно сделать и таким образом, но группа «флажков» была бы, вероятно, проще и более интуитивна. Вкладка «Options» дает быстрый доступ к параметрам настройки, например, скорости выгрузки/скачивания и ограничению соединений. Они мгновенно вступают в силу, и кроме того, доступны в контекстном меню (по нажатию правой клавиши мыши) на главной панели «торрентов».

› *BitTorrent* жадноват до ресурсов, но его просто нельзя не попробовать!

Умные функции

Azureus интуитивно понятен в работе – в зависимости от выбранного уровня использования грамотно подобрано число доступных опций настройки, чтобы приложение не выходило за рамки заданной сложности. Частично загруженные «торренты» из других клиентов легко импортируются для возобновления загрузки или продолжения их раздачи с помощью *Azureus*. Аналогично другим клиентам, это приложение при следующем запуске возобновляет работу со всеми предыдущими «торрентами», а кроме того, предлагает собственную функцию «Fast Resume». Хотя нормальное возобновление работает только с завершенными фрагментами «торрента», *Azureus* способен сохранять частично загруженные фрагменты до следующего запуска, когда он перезапустится с середины фрагмента.

Поддержка закачек – дело важное, но не менее важно изобилие дополнительных функций. Создать новый торрент-файл поразительно легко: просто выберите файл или каталог для публикации и присвойте трекеру имя. Другая приятная функция – способность размещать трекер прямо в *Azureus*, что идеально для распространения частных «торрентов» среди небольшой группы людей, не нуждающихся в публичном трекере. Как и многие другие клиенты, *Azureus* поддерживает децентрализованный *BitTorrent* с помощью разновидности DHT (распределенных хэш-таблицы), однако его тип DHT отличается от основного клиента *BitTorrent* и годится только для пользователей *Azureus*. Правда, таких пользователей полно, но трудно игнорировать тот факт, что обособленный подход к DHT делает *Azureus* чуть менее привлекательным.

Наконец, какое приложение в наши дни считается завершенным без способности к расширениям? Если расширять любимые приложения – ваш конек, то *Azureus* сослужит вам хорошую службу. Несколько расширений поставляются в стандартной инсталляции, но беглый просмотр страницы **http://azureus.sourceforge.net/plugin_list.php** обнаруживает массу дополнительных, от внешнего вида до методов удаленного доступа. Из наиболее ярких – отправка сообщений о статусе по электронной почте и SMS; личности вроде нас хотят сразу же узнавать, что закачка дистрибутива завершена, без лихорадочной ежеминутной проверки статуса!

Клиенты на базе Java

Поскольку приложения Java обычно компилируются в байт-код, требуется, чтобы до запуска приложения была развернута виртуальная машина. Она предоставляет независимость от платформы, но ценой общей производительности. Вы можете выбрать одну из нескольких виртуальных машин на выбор, включая Hot Spot от Sun и пару GNU-альтернатив.

К сожалению, нам не удалось запустить *Azureus* под GNU-интерпретатором Java (*GIJ*). Для компиляции Azureus в родной байт-код при помощи GNU-компилятора Java (*GCJ*) было сделано многое, но на данном этапе советуем для верности установить последнюю JRE от Sun.

«Поддержка закачек – это важно, но не менее важны другие функции.»

LINUX FORMAT Вердикт

Azureus

Версия: 2.5.0.0
Сайт: http://azureus.sourceforge.net
Цена: бесплатно по лицензий GPL

» Здесь достаточно функций и опций, чтобы удовлетворить любого, а с расширениями его возможности действительно безграничны.

Рейтинг 9/10

BitTornado

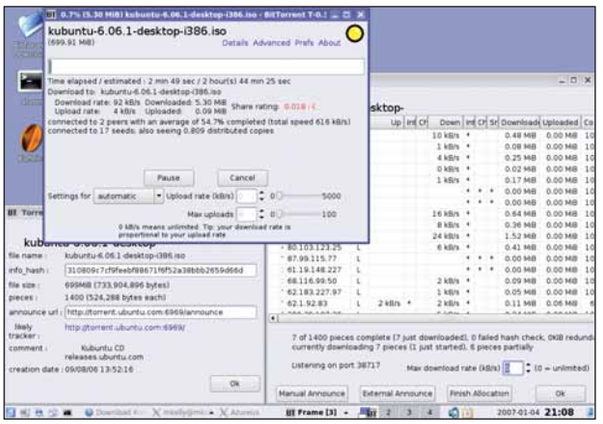
Расширенная версия основного клиента.

Пользователи более старых официальных клиентов *BitTorrent* мгновенно разберутся, как работает *BitTornado* – это, по сути дела, параллельная ветвь первоначального клиента; на самом деле, дистрибутивы на основе Debian, как правило, ставят два набора ПО параллельно и проецируют один набор для использования под общими именами программ. Как можно ожидать, *BitTornado* поставляется с набором отдельных программ, которые решают задачи типа скачивания конкретного «торрента» при помощи графического интерфейса, консоли или curses, а также запуска множественных загрузок одной командной строкой. Есть еще инструменты для создания торрент-файлов и манипулирования ими, а заодно и трекером.

Впервые появившись, *BitTornado* заявил ряд улучшений по сравнению с официальным клиентом, хотя справедливо будет отметить, что разработка официального клиента с тех пор все наверстала. Среди разрекламированных функций – поддержка UPnP и назначение приоритетов отдельным файлам. К сожалению, мы не сумели добиться работы UPnP, и пришлось прибегнуть к ручному пробросу портов, но людям этот фокус удавался. Одно из основных ограничений – невозможность управлять множеством «торрентов» в одном графическом окне. Другие клиенты уже управляют множественными соединениями, а в *BitTornado* приходится делать все вручную, путаясь в толпе открытых на экране окон. Можно использовать один из консольных инструментов, разработанных для множественных загрузок, но они довольно топорны и не интерактивны.

Неплохо, что *BitTornado* требует мало памяти и прост в работе, но этого мало, чтобы оправдать его использование. Заслуженное признание он получает как основа для других *BitTorrent*-менеджеров, типа *TorrentFlux*, или как компонент пользовательских скриптов.

> «Пока невозможно управлять множеством «торрентов» из одного графического окна.»



› ***BitTornado*** **дает полезную информацию и множество опций, но пока не развит как клиент.**

LINUX FORMAT Вердикт

BitTornado

Версия: 0.3.17
Сайт: www.bittornado.com
Цена: бесплатно по лицензии MIT

» *Полезный инструмент как основа для другого ПО, но не внушающий уважения при непосредственном использовании.*

Рейтинг 4/10

KTorrent

Собственный торрент-клиент KDE.

Возможно, вы не согласитесь с нами, но мы нашли, что приложения, ориентированные на KDE или Gnome, имеют тенденцию быть бледными копиями автономных версий. К счастью, *KTorrent*, похоже, имеет большинство свойств, предлагаемых лучшими клиентами, и поддерживает большую часть требуемых функций, хотя и не без посторонней помощи.

Работает это приложение вполне предсказуемо – списки скачиваемых и раздаваемых «торрентов» появляются вверху, а пара вкладок расположена внизу экрана. Вкладки выводят данные о выбранном торренте, в списке файлов помечаются выбранные для загрузки. К сожалению, без расширения infowidget доступна весьма скудная информация.

И это ключевой момент касательно *KTorrent*: будучи просто установлен в конфигурации по умолчанию, он функционально бедноват и требует активации ряда расширений через меню настройки. Со стандартной инсталляцией идут расширения, предоставляющие импорт торрент-файлов, UPnP и IP-фильтрацию. Приятно, что они существуют, но в наши дни данные функции почти обязательны для подобного приложения, и не должны бы выступать как опциональное дополнение.

Подключены расширения или нет, но в некоторых областях *KTorrent* явно слаб. Мы не сумели найти способа точной настройки приоритета определенного «торрента» по сравнению с другими – здесь есть менеджер очередей, но зачем он, не ясно, поскольку документации к нему не имеется. И хотя можно назначать приоритеты отдельным файлам, нету указаний, применимы ли они к множественным «торрентам». В принципе, приложение стоящее, и его включили в состав рабочего стола KDE не зря. Мы использовали последний стабильный релиз; возможно, версия 2.1 включит больше функций.

> «Вынутый из коробки KTorrent бедноват.»

› **Функции, необходимые любому *BitTorrent*-клиенту, упрятаны в Раздел настройки расширений.**

LINUX FORMAT Вердикт

KTorrent

Версия: 2.0.3
Сайт: http://ktorrent.org
Цена: бесплатно по лицензии GPL

» *Полезный и стабильный клиент, но побольше бы стандартных функций.*

Рейтинг 6/10

TorrentFlux

Многопользовательский торрент-менеджер на базе web-технологий.

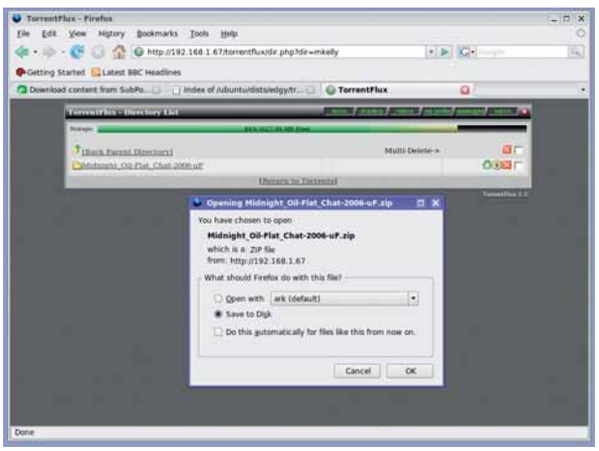
Не каждый любитель скачивать «торренты» заставляет полноценный графический клиент работать круглосуточно. Позвольте представить *TorrentFlux*, обещающий web-интерфейс для закачек на машине без постоянного контроля: он идеален для домашних серверов, подключенных к широкополосному каналу, или для освоения лимита трафика на учетной записи хостинга.

Мы мигом привели это приложение в рабочее состояние из архива, но помните о зависимостях: понадобится *Apache*/PHP, *MySQL* и как минимум Python 2.2, поскольку за сценой *TorrentFlux* скрывается *BitTornado*. *BitTornado* поддерживает UPnP, и нас малость удивило, что в *TorrentFlux* это, похоже, не работает: пришлось открывать порты на маршрутизаторе вручную. К сожалению, здесь выплыла еще одна проблема: отсутствие документации. Опытные пользователи *BitTorrent*-клиентов разберутся с системой довольно быстро, но новичкам, пожалуй, лучше сперва поупражняться со специализированным клиентом.

Необычная функция *TorrentFlux* – многопользовательский аспект. При первом входе создается пользователь-администратор, имеющий право создавать последующих пользователей. Каждый пользователь может затем добавлять свои собственные «торренты» для загрузки и, как только «торрент» скачан, забирать его из своей области. Это позволяет администратору отслеживать такие аспекты, как занятость канала, и ограничивать число «торрентов» с помощью менеджера очереди, чтобы пользователи не заграбастали себе его весь.

TorrentFlux – отличный маленький клиент, хотя в некоторых отношениях пока несовершенен. Например, можно создавать «торрент» только из файлов, уже расположенных в области закачки, так что создание новых «торрентов», безусловно, несколько затруднено. Web-интерфейс имеет свои причуды – все выполняется не столь мгновенно, как в нормальном клиенте.

> «Администратор отслеживает занятость канала.»



› **Для упрощения скачивания нескольких файлов за один раз, *TorrentFlux* пакует их в архив – удобно!**

LINUX FORMAT Вердикт

TorrentFlux

Версия: 2.3 final
Сайт: www.torrentflux.com
Цена: бесплатно по лицензии GPL

» *Когда требуется удаленный доступ, это верное дело.*

Рейтинг 8/10

Transmission

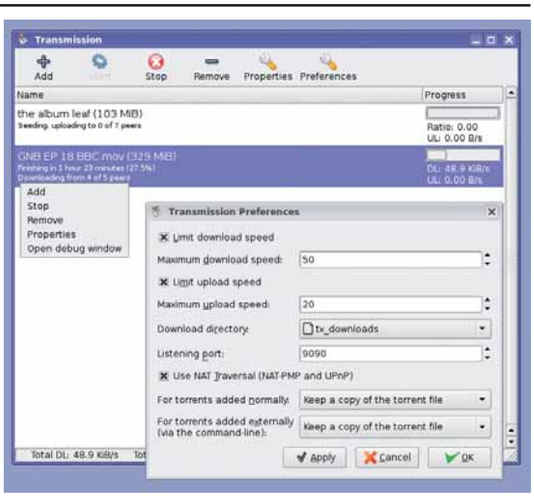
Легковесный клиент для запуска ваших закачек.

Transmission – это кроссплатформенный клиент, работающий на Mac OS X, BeOS и Linux с помощью инструментария *GTK*. Сами же разработчики называют его легковесным, и это ясно с первого запуска: вы получаете всего лишь панель инструментов и список «торрентов». *Transmission* весьма похож на графический интерфейс официального клиента *BitTorrent*, только без информации внизу о выделенном в данный момент «торренте».

Минималистический интерфейс, вероятно, обусловлен наследием приложений Mac OS X – следует признать, что экранные снимки Mac-версии смотрятся неплохо. Тем не менее, есть разница между минималистическим интерфейсом и минимальной функциональностью, и в интерфейсе *Transmission* все наружу. Кнопки на панели инструментов используются для открытия торрент-файлов, управления существующими «торрентами» и настройки приложения. Уровень управления закачками, мягко говоря, невысок. «Торрент» можно только остановить или удалить – нет никакого способа просмотреть детальную информацию о содержимом файлов или задать приоритет одного «торрента» перед другим. Параметры настройки сводятся к ограничению скоростей выгрузки и скачивания и указанию прослушиваемого порта. Используя это приложение, вы не сможете управлять вашими «торрентами» должным образом.

Еще хуже то, что ряд приватных трекеров не допускают его к использованию своих сервисов из-за «злоупотребления протоколом». Мы не эксперты в тонкостях спецификаций *BitTorrent*, но *Transmission*, видимо, играет не по правилам, а разработчики не спешат это починить. К сожалению, похоже, что *Transmission* зажился на свете просто из-за недостатка конкурентов на Mac OS X, а *GTK*-версия – лишь побочный продукт. Единственно за свои заслуги, в Linux он быстро оказался бы на свалке.

> «Уровень управления закачками, мягко говоря, невысок.»



› **Этот рисунок воплощает почти всю функциональность, которую предоставляет *Transmission*. В самом деле!**

LINUX FORMAT Вердикт

Transmission

Версия: 0.6.1
Сайт: http://transmission.m0k.org
Цена: бесплатно по лицензии MIT

» *Минималистический интерфейс пошел бы мощному клиенту, но здесь просто мало функций.*

Рейтинг 4/10

Клиенты BitTorrent

Вердикт

Azureus 9/10

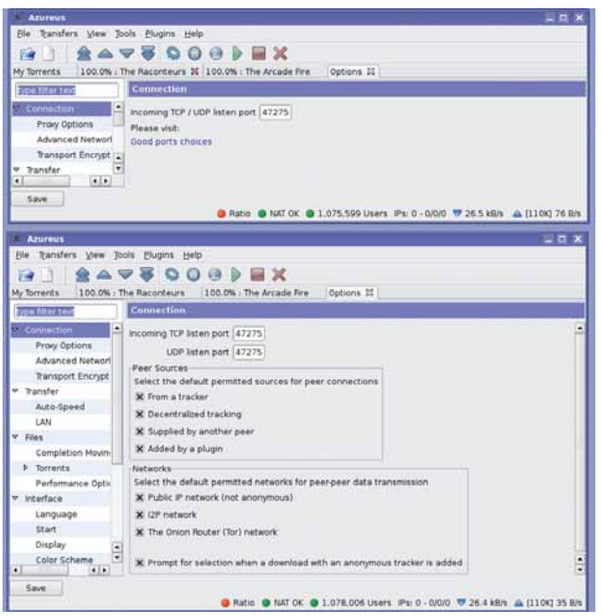
Мы дали довольно строгие оценки вполне приличным *BitTorrent*-клиентам. Мы уже видели клиент *BitTorrent*, встроенный в web-браузер Opera, так что справедливости ради нужно отметить, что сам по себе протокол становится обычным делом, поэтому мы не рассматривали стек протокола как таковой, а разбирались с тем, что доступно вокруг протокола с точки зрения удобства управления. И более всего нас заботило, насколько хорошо клиенты позволяют управлять множественными «торрентами». Люди используют *BitTorrent* все чаще и чаще, и не только для загрузки дистрибутивов Linux. Речь уже не идет о «чистой» скорости или лучшей реализации протокола... в фокусе оказывается способность жонглировать текущими закачками и скоростями, с целью выжать все из своего соединения.

И поэтому *Azureus* победил. Никакой другой клиент не предлагает столько гибкости в столь мощном пакете. Хотя многие считают, что *Azureus* жаден до ресурсов, временами приходится платить эту цену ради лучшей из программ. Он содержит гору функций, дружелюбен к пользователю, и его базовая функциональность, и без того впечатляющая, умножается через интерфейс расширений.

Удивительно сильную оппозицию для *Azureus* составил *TorrentFlux*. Со точки зрения «голых» функций до *Azureus* ему далеко, но если вам нужен торрент-менеджер, доступный удаленно и способный запускаться в неграфическом окружении, лучше, чем *TorrentFlux*, вы не найдете. Тот факт, что он предлагает многопользовательскую функциональность, также произвел впечатление, и мы отметили появление учебников в интернете, рассказывающих, как приспособить «машину с *TorrentFlux*» в качестве выделенного сервера закачек.

Когда у вас в лидерах два таких замечательных приложения, трудно говорить об остальных претендентах. *KTorrent* и *Deluge* имеют похожее происхождение и выглядят очень многообещающе, но оба страдают от одной и той же проблемы: недостатка функций и документации. Чтобы быть справедливыми к разработчикам *Deluge*, отметим, что это довольно молодой проект, но, наверно, им стоит попридержать новые релизы, пока не будет добавлено чуть больше функций.

А насчет официального клиента *BitTorrent* – ну, надо постараться, чтобы занять столь видную позицию, создав себе имя и .com-адрес, и затем все это слить, выпустив средненький продукт; но его авторы умудрились сделать это. Лучше бы они совершенствовали протокол и свой портал, а функциональность своего клиента оставили на экспертов.



› *Azureus* показывает одну и ту же страницу настройки для пользователей с различным уровнем подготовки – обратите внимание на простоту верхней панели. LXF

Обратная связь

Приведет ли открытие кода JVM от Sun к улучшению производительности *Azureus*? Нагонит ли документация когда-нибудь разработку приложений? Преуспеет ли BBC в распространении старых шоу через BitTorrent? Напишите свое мнение на letters@linuxformat.ru.

> «Люди используют BitTorrent все чаще, и не только для загрузки дистрибутивов Linux.»

Таблица функций

Клиент	Ограничение скорости	Поддаержка UPnP	Импорт торрентов	DHT[1]	Уведомления	Встроенный трекер	Возобновление	Предпросмотр файлов	Частичные загрузки	Приоритеты [2]	Использование памяти [3]	Использование процессора [3]	Создание торрентов
Azureus	✓	✓	✓	Собственный	Плагин	✓	✓	✓	✓	Файлы	Плагин	Плагин	✓
TorrentFlux	✓	☒	☒	☒	☒	☒	✓	✓	✓	Торрент	✓	✓	✓
KTorrent	✓	✓	✓	Основной	☒	☒	✓	✓	✓	Файлы	☒	☒	✓
BitTorrent	✓	✓	✓	Основной	☒	Предоставляется	✓	☒	☒	Файлы	☒	☒	✓
BitTornado	✓	✓	✓	☒	☒	Предоставляется	✓	☒	✓	Нет	☒	☒	✓
Deluge	✓	☒	✓	Основной	☒	☒	✓	☒	✓	Торрент	☒	Плагин	☒
Transmission	✓	✓	✓	☒	☒	☒	✓	☒	☒	Нет	☒	☒	☒

[1] «Собственный» – это собственная реализация DHT в Azureus. «Основной» – реализованный в официальном клиенте. [2] Варианты: назначение приоритетов файлам, «торрентам» и отсутствие приоритетов вообще. [3] Может предоставляться дополнительным плагином..

Distrowatch

Ежемесячная сводка новостей дистрибутивов Linux

ЛАДИСЛАВ БОДНАР
основатель, начальник, редактор и сотрудник DistroWatch.com.

Модная шляпка

Вы интересуетесь развитием Fedora? Если ваш ответ: «нет, у меня другой дистрибутив, и разработки Red Hat меня не волнуют», то вы лишаете себя удовольствия: ведь Fedora – законодатель мод современных дистрибутивов Linux!

Позвольте привести несколько примеров. Помните времена, когда Red Hat ввела загрузчик *Grub* в Red Hat Linux 7.2? Лишь немногие дистрибутивы рискнули использовать этот необкатанный альфа-релиз, но очень скоро большинство последовало за Red Hat и отказалось от *Lilo*. Точно так же ext3 стала самой популярной файловой системой не потому, что была лучшей, а потому, что Red Hat избрала ее единственной поддерживаемой журнальной файловой системой для своих продуктов. И таких историй множество.

Финансовый мускул

Лидирующая позиция Red Hat в инновациях Linux не есть результат случайности. Как фирма, построившая успешный бизнес на Linux, она обладает и достаточной властью, и наличными, чтобы влиять на разработку жизненно важных компонентов Linux: именно она платит Алану Коксу [Alan Cox] за работу над ядром Linux, она вкладывается в разработку *glibc* и *GCC*, и она поддерживает LVM, Logical Volume Manager. Поскольку все эти программы выходят под лицензией GPL и используются в других дистрибутивах, идеи Red Hat затрагивают всех нас, независимо от того, каким дистрибутивом мы пользуемся.

Ах, да, есть и еще одна причина следить за развитием Fedora – это жутко интересно. Многие дистрибутивы Linux выдвигают интересные идеи, но лишь у немногих из них хватает финансовых ресурсов, чтобы воплотить их в жизнь.

ladislav.bodnar@futurenet.co.uk

Маневр Fedora

Fedora 7 ‘Core’ больше нет! Мало того: на встрече разработчиков объявлено о смене направления проекта сообщества Red Hat.

Вестфорд, Массачусетс, конец 2006 г. Небольшая группа влиятельных разработчиков Fedora собирается, чтобы обсудить будущее дистрибутива и планы на следующий релиз. На первый взгляд – обычная встреча, чтобы обсудить, насколько продвинулся проект с октябрьского выхода Fedora Core 6. Но к концу встречи стало ясно, что это – отнюдь не ординарное рандеву, а собрание, которому суждено иметь далеко идущие последствия для всех пользователей Fedora.

Mirin Duffy

› Семерка будет даром Небес? ‘Fedora Borealis’ – лишь одна из тем рабочего стола, созданных художниками Fedora.

Когда принятые решения обнародовали, многие фанаты Fedora были потрясены масштабом изменений, предстоящих их любимому дистрибутиву. Разработчики хотят не просто сделать свой проект новаторским и неординарным; они готовы поучиться успеху у других популярных дистрибутивов, в особенности Ubuntu. В конце концов, Ubuntu стал самым быстроразвивающимся настольным дистрибутивом Linux и особенно преуспел в привлечении новых пользователей – а Fedora по этой части отстает, несмотря на то, что существует дольше.

Показатель серьезного отношения к конкуренции – планируемое разветвление Fedora 7 на несколько дистрибутивов целевого назначения (а-ля Ubuntu): Fedora Desktop, Fedora Server и Fedora KDE. Вдобавок, нынешнее различие между базовыми пакетами («core») и расширенными («extras») будет устранено, и всё ПО целиком сольётся в один огромный репозитарий, содержащий более 3 600 RPM. Слияние потребует большой работы, как сообщил в январе участникам списка рассылки разработчиков Fedora Билл Ноттингем [Bill Nottingham] из Red Hat. «Оставшиеся члены Core Steering Committee поработают с Советом директоров Fedora и FESCO (Extras Steering Committee), и выяснит, как управляться с этим новый комбинированным репозиторием», написал он.

> «Фанаты Fedora были потрясены масштабом изменений.»

Наиболее часто критикуемая область Fedora – это инфраструктура управления пакетами: печально известная медлительность *Yum* (в особенности по сравнению с *apt-get* от Debian) и отсутствие графического интерфейса пользователя. Пока непохоже, что Fedora 7 включит *Synaptic*-подобную утилиту, которая позволит пользователям легко добавлять в *Yum* сторонние репозитории, но имеются четкие планы увеличить скорость инструмента управления пакетами на основе Python, отыскав и устранив его «узкие места».

В Fedora 7 также ожидаются увеличение скорости загрузки (с последующим анализом альтернатив для *init*), отлаженная поддержка «из коробки» всех поддерживаемых беспроводных сетевых карт (включая шифрование) и шифрованные файловые системы. Обещана также улучшенная поддержка ноутбуков, особенно в сфере управления питанием; еще мы, должно быть, увидим различные усовершенствования рабочего стола, например, переключение между пользователями и новую версию *RandR* – расширения *X.org*, которое позволит динамически изменять размеры экрана и поворачивать его, и прочие современные функции.

Выход Fedora 7 намечен на 24 апреля 2007 г. [и уже перенесен на 24 мая, – прим. ред.] Правда, учитывая масштаб изменений и исторически сложившуюся неспособность разработчиков Fedora соблюдать намеченные сроки выпуска, ждите задержки прибытия. Но похоже, что Fedora 7 оправдает ожидания.

http://fedoraproject.org

Турецкие сласти

Pardus Linux 2007 Лощеный настольный дистрибутив с наглядным управлением пакетами.

Вы по-прежнему в поиске идеального дистрибутива для рабочего стола? Тогда не пропустите Pardus Linux 2007. Возможно, он и не принадлежит к фаворитам основных ИТ-публикаций, и у него явно не хватает средств обвешать сайты Linux своей рекламой, но этот турецкий дистрибутив содержит достаточное количество инноваций и достаточно хорошо отшлифован, чтобы стать одним из лучших новых дистрибутивов Linux 2007 года.

› Pardus интегрировал стартовое меню *Kickoff* от OpenSUSE.

Проект начинался как ответвление Gentoo Linux, но вскоре разработчики отошли от родительского дистрибутива и занялись разработкой собственных решений. Одним из них было включение в дистрибутив новой системы управления пакетами под названием *PiSi* – решение на удивление хорошее, предлагающее отличные инструменты командной строки, полнофункциональный графический менеджер пакетов, утилиты уведомления об обновлениях и солидную инфраструктуру для получения и обновления дополнительных пакетов ПО из репозитариев с хорошей поддержкой.

Еще одно настраиваемое приложение, которое порадует многих пользователей Linux – центр настройки Tasma, это утилита, пошедшая путём *Yast* от OpenSUSE или *Mandriva Control Center* и предоставляющая легкий доступ ко многим опциям рабочего стола и модулям настройки KDE.

А ещё разработчики Pardus Linux создали массу менее универсальных утилит: например, *Comar* – менеджер настройки, или *Mudur* – замена *init*. Но самое сильное впечатление, конечно, производят отличный менеджер пакетов и быстрый и симпатичный рабочий стол.

www.pardus.org.tr/eng

Ради жизни Linux

SystemRescueCd 0.3 Восстановление диска и данных в удобном и компактном дистрибутиве.

SystemRescueCd – одна из тех жемчужин, без которых не обойтись ни системному администратору, ни продвинутому пользователю. Невыразительное название («системный спасатель») приводит на ум крутых хакеров, набирающих в черном терминале загадочные команды, но на самом деле SystemRescueCd – современная графическая утилита, пригодная для решения самых разных задач: от разбиения жесткого диска на разделы до исправления ошибок файловой системы и полного клонирования жестких дисков.

› *WindowMaker* – быстрый, элегантный рабочий стол SystemRescueCd.

Основной инструмент на CD – *GParted*, графическая утилита разбиения на разделы, она поддерживает большее количество файловых систем, чем любой другой аналогичный инструмент из ныне существующих: диск можно разбить на разделы аж с 13-ю разными файловыми системами, а заодно поддерживается перемещение разделов и изменение их размеров – и все это совмещено с удобством интуитивного графического интерфейса.

Последний релиз SystemRescueCd содержит немало новых полезных функций: например, поддержку чтения/записи для файловой системы NTFS и антивирус *ClamAV*. Кроме того, на CD есть несколько текстовых редакторов и web-браузеров, а большинство низкоуровневых дисковых утилит обновлены до последних версий. Помимо самого CD, проект, основанный в 2003г. Франсуа Дюпу [Francois Dupoux], поддерживает сайт в стиле Wiki с превосходной документацией.

SystemRescueCd отлично борется с такой малоприятной проблемой, как восстановление файлов, а эта проблема периодически лягает большинство пользователей. Держите его под рукой, чтобы встретить неприятность во всеоружии.

www.sysresccd.org

Опасные уровни активности!

Один из способов определить популярность дистрибутива – проверка активности его онлайн-сообщества, например, форумов. В приведенной ниже таблице мы приводим количество пользователей на форумах нескольких дистрибутивов. Как видите, удивительный рост популярности Ubuntu отражен числом участников UbuntuForums.org, более чем вдвое превышающим число участников форума ближайшего из дистрибутивов-конкурентов. Но, пожалуйста, не забудьте, что не у всех дистрибутивов есть онлайн-форумы: например, Debian предпочитает списки рассылки.

Дистрибутив	Число пользователей на форумах	Форум
Ubuntu	215,200	http://ubuntuforums.org
Gentoo	103,000	http://forums.gentoo.org
Fedora	84,500	http://forums.fedoraforum.org
Mandriva	54,800	http://forum.mandrivaclub.com, http://mandrivausers.org
OpenSUSE	41,800	www.suseforums.net, http://forums.suselinuxsupport.de
Xandros	24,200	http://forums.xandros.com
Kubuntu	14,000	http://kubuntuforums.net
Freespire	10,100	http://forum.freespire.org
Arch Linux	7,500	http://bbs.archlinux.org
PC-BSD	5,700	http://forums.pcbsd.org
PCLinuxOS	4,800	www.pclinuxos.com/forum

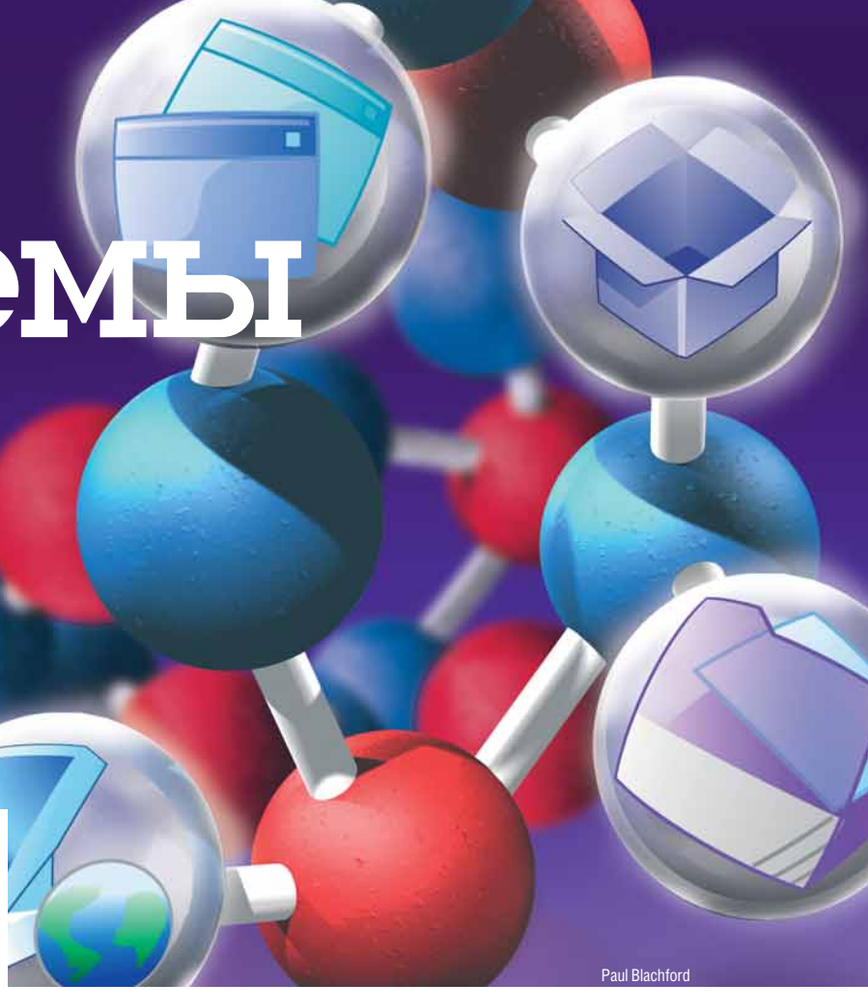
Хит-парад дистрибутивов

10 самых посещаемых страниц на DistroWatch.com за четыре недели с 01/08/06 по 29/08/06 (среднее число визитов в день)

Дистрибутив	Количество визитов	
1 **SUSE**	2,220	‹›
2 **Ubuntu**	2,170	‹›
3 **Fedora**	1,261	‹›
4 **SimplyMepis**	1,148	↑
5 **Debian GNU/Linux**	975	↑
6 **Sabayon Linux**	954	↑
7 **PCLinuxOS**	894	‹›
8 **Mandriva**	822	↓
9 **Knoppix**	766	↑
10 **Damn Small Linux**	738	↓

DistroWatch.com отслеживает популярность дистрибутивов, основываясь на количестве посещений страницы, посвященной данному дистрибутиву. Хотя эти цифры не отражают реального количества инсталляций, они все же являются индикатором популярности дистрибутива в определенный период времени.

Проблемы Linux РЕШЕНЫ



Что, завязли? Руководство **Грэма Моррисона** по типичным проблемам в Linux должно стронуть вас с места.

Paul Blachford

Где найти свои вопросы:

Установка с. 26

Оборудование с. 28

Программное обеспечение с. 30

Сети с. 32

Администрирование с. 32

Нечего стыдиться, если вы застряли на простой с виду проблеме в Linux. Знайте: множество вопросов по Linux люди задают снова и снова – и не потому, что ленятся искать ответы самостоятельно, а потому, что очень часто на это нет времени или данное проявление проблемы не похоже на обычное. Почему видеодрайверы так трудно устанавливать? Что случилось с моим интернет-соединением? Почему не устанавливается эта программа?

Linux не является причиной проблем. В наши дни можно твердо сказать, что с Linux проблем не больше, чем с любой другой операционной системой, и подавляющее их большинство – это результат почти бесконечных комбинаций оборудования, которые можно встретить на типичных ПК. Обычный источник проблем – плохо поддерживаемое оборудование или то, что пользователь не знаком с ОС. Для большинства пользователей установка диска с последней версией Ubuntu в CD- или DVD-привод окончится без неприятностей. Но некоторые общие проблемы время от времени выплывают наружу – будь то «железо» или ПО, установка или администрирование. И из-за многовариантной природы Linux и оборудования, на котором он работает, бывает трудно точно указать причину.

На последующих страницах – кроме подробных ответов по типовым проблемам – мы еще и попытаемся научить вас решать такие проблемы. Ключ к решению – понимание взаимодействия ОС и оборудования, а также работы различных компонентов вашего дистрибутива Linux. Глубокое знание Linux само по себе награда, но оно также помогает разобраться с неприятными инцидентами.

Мы должны предупредить вас, что – хотя это и очевидно – некоторые внесенные изменения могут иметь катастрофические последствия для вашей Linux-системы, если они сделаны неправильно. Поэтому обязательно обезопасьте все ваши данные: вдруг что-то пойдет наперекосяк. Нам всем случалось раскаиваться, что мы отложили резервное копирование на потом; итак, вашим первым заданием, еще до чтения, будет создание резервной копии!

Установка

Что делать, если все отказало еще до того, как дистрибутив оказался на вашем жестком диске.

У некоторых невезучих Linux даже не устанавливается. Вы-то думали, что инсталляция – самый отлаженный аспект дистрибутивов Linux, поскольку и разработчики, и пользователи должны пройти одни и те же этапы; но это, к сожалению, неверно.

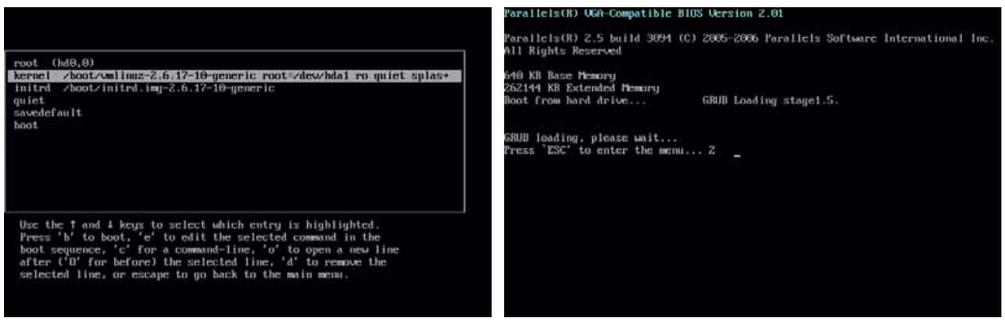
Последние релизы, как Fedora Core, так и Ubuntu, намучились с процедурами установки, и хотя большинство пользователей запускает Linux без сучка без задоринки, итогом этого процесса может стать пустой экран и голый диск. Причины проблем весьма различны, но большинство из них сводится к оборудованию. Ваши шансы на успех сильно возрастут, если еще до инсталляции дистрибутива вы постараетесь сделать свою машину как можно более стандартной.

Это означает удаление всех ненужных для инсталляции устройств. Может помочь попытка использовать IDE-диск вместо более современного SATA-варианта, а также отключение ненужных карт расширения и USB-периферии. Linux отлично адаптируется, и устройства легко будет добавить после завершения процесса первоначальной установки.

Следует также убедиться, что вы используете последнюю версию BIOS для вашей материнской платы (ее можно скачать с сайта изготовителя вашей «материнки»). Номер версии BIOS мелькает на экране при включении компьютера. Его также можно проверить в меню BIOS, нажав [F2] или [F12] во время загрузки.

Прощай, ACPI!

Если проблем избежать не удалось, существует довольно большой шанс заставить вашу систему работать, отключив ACPI. Это функция управления питанием, встроенная в каждый современный BIOS/материнскую плату, но ее реализация далека от стандарта и может вызывать проблемы. Если вы хотите попробовать отключить ACPI, сделайте это из меню загрузки до того, как что-нибудь загрузится. Во-первых, большинство современных дистрибутивов Linux используют систему загрузочного меню, именуемую *Grub*. Установлен ли *Grub*, можно узнать по появлению сообщения «Grub loading, please wait» после инициализации BIOS. Если вы такого не увидите, ваш дистрибутив, вероятно, использует старую альтернативу, именуемую *Lilo* – попробуйте более новый дистрибутив.



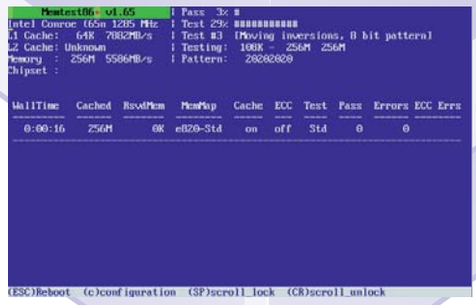
› Загрузчик *Grub* появляется при запуске Linux-дистрибутива. Можно отредактировать пункты меню и добавить более совместимые с вашим «железом» опции: для входа в режим редактирования нажмите <Escape>, а затем <e>.

Увидев сообщение «Grub», нажмите клавишу Escape. Откроется меню опций для загрузки вашей системы. Нужно выделить опцию загрузки по умолчанию (обычно вверху списка) и нажать клавишу e. Далее, выделите строку, начинающуюся со слова «kernel», и снова нажмите e. Это редактирование строки конфигурации ядра для меню *Grub*: нужно добавить «acpi=off», нажать Enter, а затем b для загрузки вашего модифицированного пункта. Если это решит вашу проблему, можете сделать данное изменение постоянным, добавив «acpi=off» в той же строке ядра в **/boot/grub/menu.lst** в вашей установленной Linux-системе. Чтобы сохранить изменения на диске, вам нужно будет отредактировать этот файл с правами администратора системы.

Ошибки памяти

Ошибку вызывать может также какой-либо аппаратный компонент. Linux предъявляет к оборудованию гораздо больше требований, чем другие ОС, и поэтому первоначальная установка Linux может первой выявить признаки сбоя. Обычно главный виновник – память, т.к. при превосходном функционировании всего модуля сбои может давать одна небольшая секция. Linux старается использовать столько памяти, сколько может потребоваться для оптимизации дискового кэша или временного файла, и прихватывает ту память, которой другие операционные системы касаться не рискуют. Ошибки памяти могут возникать случайным образом и приводить к зависанию вашей машины, повреждению изображения на экране или возникновению различных случайных системных ошибок.

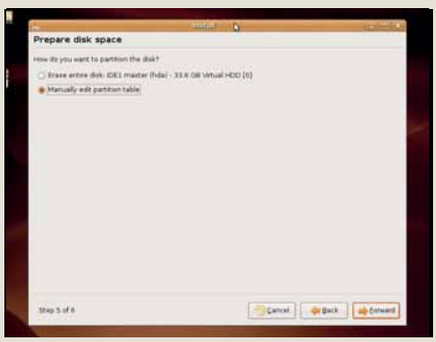
Вы можете выяснить, в памяти ли дело, запустив программу под названием *Memtest86*. Она существует уже довольно долго (она унаследована от ранней версии ядра Linux). Ее нужно загружать прямо с CD/DVD-ROM или дискеты (еще помните такие?); загрузившись, она запустит исчерпывающую серию тестов вашей памяти. Если ваша память переживет эту пытку, не выдав никаких ошибок, можете быть почти уверены, что проблем с нею нет. Если *Memtest* что-нибудь обнаружит, придется заменить неисправный модуль (купите, одолжите или стяните другой). Стоит это примерно 2-3 тыс. рублей за 1 ГБ-модуль, но вы обнаружите, что такая замена сделает более стабильными и другие ОС на этой машине. »



› Если память выдюжила испытание огнем *Memtest86*, значит, проблемы не здесь.

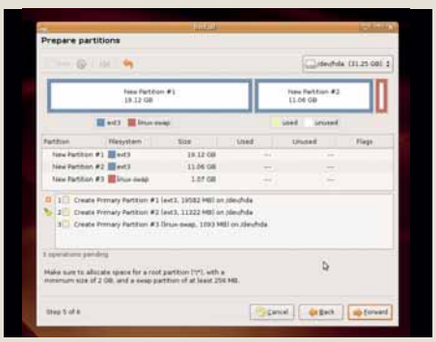
Шаг за шагом: Создаем home-раздел

Одной из наиболее полезных корректив при установке Linux-системы является создание отдельного дискового раздела для домашних каталогов. Не жалейте затраченного времени: это упростит и обновление дистрибутива, и создание резервных копий ваших данных. Системные файлы храните в корневом разделе, а пользовательские данные – в home-разделе. Во время обновления убедитесь, что инсталлятор переформатирует только корневой раздел, а home-раздел не трогает.



» Ручное редактирование

В этом примере мы воспользовались Ubuntu, но в большинстве дистрибутивов процесс похожий. Первый шаг – позаботиться о ручном изменении таблицы разделов, не позволив инсталлятору полностью управлять выделением дискового пространства.



» Выделение памяти

Как минимум, не считая Windows-раздела, вам потребуются разделы 'root', 'home' и 'swap'. Последний может быть небольшим (два объема ОЗУ), а размер остальных двух зависит от того, где вы собираетесь хранить свои данные.

» Выбор места

Наконец, укажите подходящую точку монтирования (место в вашей файловой системе) для каждого раздела: **/** для корневого раздела, **/home** для домашних каталогов и **swap** для оставшегося раздела.

Оборудование

Заставить оборудование работать – не обязательно сложная задача.

Заставить определенные «железки» работать в Linux пока что проблема. На это есть две принципиальных причины. Во-первых, возможных комбинаций оборудования бывает множество. Типичная ситуация – когда материнская плата от одного изготовителя, видеокарта – от другого, а внешняя периферия, например, принтер, мышь и клавиатура – от толпы каких-то других сборщиков. И каждому подавай свой драйвер, а драйверы часто еще и взаимодействуют друг с другом. Понятно, что изготовитель не в силах протестировать все возможные комбинации драйверов.

Это сочетается со второй причиной: многие изготовители пока отказываются открыть спецификации своего оборудования, чтобы разработчики могли писать драйверы. Linux уже может похвастать превосходной поддержкой оборудования от Intel, IBM и AMD, но по-прежнему очень важно убедиться в совместимости нового оборудования, прежде чем выкладывать за него денежки. Следует также проверить наличие открытого драйвера для вашего оборудования, поскольку сообщество мало чем поможет, если драйвер закрыт. Чтобы разобраться в этих вещах, загляните на **http://cdb.suse.de**, **www.mandriva.com/hardware** или **www.phoronix.com/lch**.

Проблемные видеокарты

Типовая «горячая тема» для оборудования в Linux – заставить работать видеокарту с 3D-ускорителем. По умолчанию большинство Linux-дистрибутивов устанавливают для оборудования открытый драйвер. Последняя версия Ubuntu может прервать эту тенденцию, включив проприетарные драйверы, но покамест вы получаете видео-драйверы с открытым кодом, обеспечивающие лишь минимальную поддержку 3D-ускорения. Этого достаточно для обычного использования настольного ПК, но если вы захотите поиграть в *PlanetPenguin Racer* или создать сцену в *Blender*, ваша видеокарта не справится с нагрузкой.

Если вы собрались купить новую 3D-видеокарту, для серьезной производительности годятся лишь два поставщика. Nvidia и ATI (ныне принадлежащая AMD) предлагают к загрузке собственные закрытые драйвера для ваших карт, и каждый использует свою процедуру установки. Альтернатива для тех, кто не гонится за модной «трехмеркой» – карты Intel; это оборудование не столь мощное, зато драйверы полностью открыты. Если карта у вас уже есть, выясните, какой чипсет в ней используется (для этого наберите в командной строке **lspci**).

Независимо от вашего оборудования, нужно будет установить исходные коды ядра. Каждый драйвер должен быть скомпилирован именно с той версией ядра, которую вы используете. Nvidia и ATI пытаются предоставлять прекомпилированные двоичные версии для нескольких популярных ядер и дистрибутивов, но поскольку именно ядро обычно первым заменяется при установке обновлений безопасности, нет гарантии, что эти версии за ним поспевают.

Исходный код ядра можно установить с помощью менеджера пакетов вашего дистрибутива, выполнив поиск либо по ‘kernel-source’ (для систем Mandriva и Fedora), либо по ‘linux-source’ (для систем Ubuntu/Debian). Менеджер пакетов выяснит, какая версия вам нужна, поскольку требуется ее идеальное соответствие вашему рабочему ядру, и вы всегда можете проверить, какая версия ядра у вас запущена, набрав в командной строке **uname -a**. Инструкции по установки драйвера Nvidia см. в нашем пошаговом руководстве, *справа*.

› Звук можно оживить, всего лишь загрузив настройки микшера и подвигав «бегунки».

После установки исходных текстов ядра, следующим шагом будет запуск для нового драйвера процедуры инсталляции, предусмотренной поставщиком, то есть скачивание двоичного пакета с сайта Nvidia или ATI. Перед запуском полученного файла нужно будет закрыть текущую сессию *X Window*. Некоторые дистрибутивы позволяют выйти в терминал, закрыв X-сессию, но с другими вам придется закрывать сессию вручную. Сделать это можно, введя как root либо **killall gdm**, либо **killall kdm**, чтобы остановить процесс менеджера входа в систему.

Более надежный способ достичь того же – переключить уровень исполнения (runlevel) с помощью команды *init*. Различные уровни исполнения настраиваются на запуск различных процессов. Например, уровень 1 (введите **init 1**) – обычно однопользовательская система в безопасном режиме, а уровень 5 (**init 5**) – ваш привычный, полноценный графический рабочий стол. На беду, здесь нет единого стандарта: различные дистрибутивы используют различные уровни исполнения. В Fedora и OpenSUSE пользователь может закрыть X-сессию, набрав **init 3**, а пользователям Debian/Ubuntu остается для переключения в безопасный режим только **init 1**.

Пустой экран

Инсталлятор ATI или Nvidia попытается изменить настройки вашего экрана для согласования с новым устройством. После установки вам нужно будет запустить **gdm** (или **kdm** для пользователей KDE) от имени root, чтобы активировать менеджер входа в систему. Если вы получите пустой экран, нужно будет либо вручную отредактировать **/etc/X11/xorg.conf**, либо запустить входящую в ваш дистрибутив утилиту настройки экрана. Учтите: **xorg.conf** не для малодушных. Это технический файл, включающий спецификации вашего оборудования, например, возможности экрана, а также установки драйверов. Вот несколько вещей, которые нужно проверить, если вы натолкнетесь на проблемы:

1 Убедитесь, что в секции Device конфигурационного файла подгружается правильный драйвер. Здесь должно быть либо «Driver ‘nvidia’», либо «Driver ‘fglrx’», в зависимости от вашего оборудования.

2 Разрешение экрана может выходить за пределы возможностей вашего монитора. В этом случае создайте безопасное разрешение,

«Скорая помощь» по оборудованию

» Введите **lspci** в окне терминала, чтобы увидеть список всех устройств, подключенных к вашей системе. Как правило, вы увидите больше информации, чем могли бы ожидать, потому что современные материнские платы зачастую объединяют несколько устройств в одно. Если ваше оборудование здесь не появилось, возможно, оно плохо подсоединено.

» Есть несколько способов растянуть ваш рабочий стол на два экрана. Стандартный метод – использование *Xinerama* и дублирование различных секций в **xorg.conf**, но в случае с оборудованием от Nvidia можно также воспользоваться инструментом от изготовителя и активировать режим «TwinView».

» Если вы хотите использовать с Linux MP3-плейер, выбирайте такой, который монтируется как внешний накопитель. Вы сможете скопировать на него свои музыкальные файлы как на USB-диск и управлять своей коллекцией при помощи обычного файлового менеджера.

» При сборке собственных приложений и драйверов в Ubuntu нужно установить базовое окружение для сборки, введя в командной строке следующее:

```
sudo apt-get install build-essential linux-headers-`uname -r`
```

» Когда вы собираете собственный модуль драйвера или устанавливаете проприетарный модуль от поставщика оборудования, вы сможете гарантировать его подгрузку во время старта системы, добавив имя модуля в файл **/etc/modules**.

оставив только 800x600 для всех элементов Display в секции Screen.

3 Драйвер ядра может не загружаться. Наберите **lsmod | grep "nvidia \| fglrx"**. Если на выходе ничего не окажется, введите либо **modprobe nvidia**, либо **modprobe fglrx**, для загрузки модуля драйвера.

После установки нового драйвера ваша Linux-система сможет поддерживать графику с 3D-ускорением.

Следим за перегревом

Когда ваша система работает под нагрузкой, вполне естественно, что ее компоненты перегреваются. Мощная видеокарта весьма способствует этой проблеме: без адекватного охлаждения могут начаться всякие странные вещи. Мы сказали «странные вещи», а не что-то более определенное, поскольку невозможно предсказать, что именно может случиться. На каком-то этапе возникают ошибки, обрушиваются приложения, повреждается изображение на экране, а то и система полностью зависает. А похоже это все на ошибки памяти.

Отследить такие ошибки трудно, потому что за время перезагрузки машины оборудование охлаждается, и система опять работает как следует. Если вы заподозрили, что причиной проблем может быть перегрев, запустите инструмент слежения за температурами системы: это хорошо даже в том случае, если проблем и нет. Вывод предупреждения, что ваша система «разгорячилась» – прекрасный способ пресечь проблемы до того, как они возникли.

Многие современные машины выводят информацию о температуре в каталог **/proc/acpi/thermal_zone/** в файловой системе. Если нет, попробуйте ввести **modprobe thermal** от имени root. Кроме того, существуют инструменты для мониторинга температуры и предупреждения о проблемах. Есть лишь один реальный претендент на роль лучшего приложения мониторинга: пакет под названием *Lm-sensors*. Мониторинг – это приложение уровня ядра, и когда *Lm-sensors* будет установлен, его собственные драйверы смогут собирать все необходимые данные.

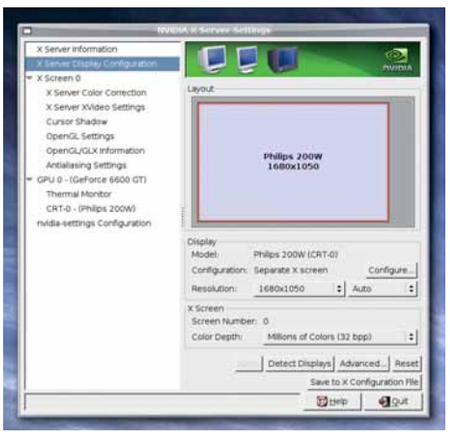
Установка обычно сводится к поиску 'lm-sensor' в менеджере пакетов вашего дистрибутива, кроме того, имеет смысл установить графическую оболочку, например, не зависящий от рабочего стола Xsensors (для KDE попробуйте *Ksensors*). Когда он будет установлен, запустите сценарий настройки из командной строки с правами root. Просто наберите **sensors-detect** и нажимайте [Enter] для каждого ответа по умолчанию. По завершении, скрипт попросит вас скопировать и вставить секцию в файл **/etc/modules**. Это обеспечит загрузку драйверов сенсоров при следующей перезагрузке системы, и вы сможете получить доступ к температурным данным, запустив **sensors** из командной строки или через одну из графических оболочек, установленных одновременно с *Lm-sensors*.

От молчания к звучанию

Еще один источник головной боли – отсутствие звука. Без сомнения, современная звуковая подсистема в Linux очень совершенна, но она также слишком сложна, и поиски первичной причины беззвучия – занятие неблагодарное. Прежде всего проверьте все таблицы. Если вы не меняли конфигурацию оборудования, очень важно удержаться от попыток расковырять провода и динамики. Если раньше все работало, значит, настройки правильные. Возня с ними, пусть даже вы уверены, что здесь-то и причина проблемы, чревата лишь новыми бедами.

Проверьте программный микшер (особенно если вы недавно модернизировали свою машину). Лучший способ это сделать – запустить какой-нибудь аудиофайл в настольном плейере и экспериментировать с уровнями различных каналов в вашем микшере. Если интерфейс командной строки в *Alsamixer* на ваш взгляд скудноват, можете установить и использовать инструмент под названием *Alsamixergui*.

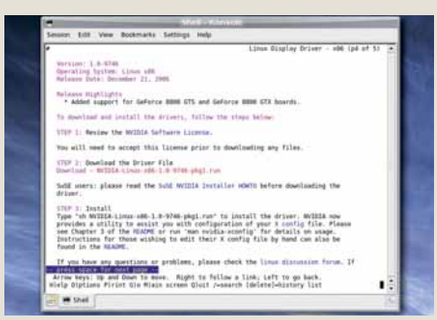
Каналы, на которые следует обратить внимание – Master, PCM и Aux, но проблема со звуком в Linux заключается в том, что функции этих каналов не всегда очевидны. Например, обычная проблема – «бегунок» IEC: чтобы звук начал воспроизводиться, иногда его нужно выставить на 25%, а иногда на 0%. Поиграйте с ним, пока что-нибудь не услышите.



› Изменить разрешение экрана можно не только редактируя файл *Xorg.conf*, но и через панель настройки Nvidia.

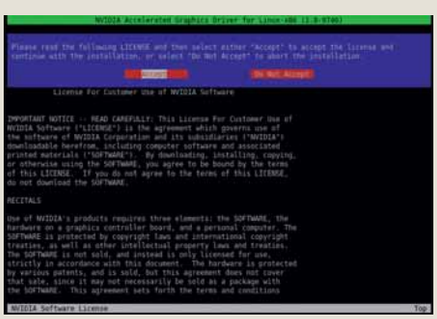
Шаг за шагом: Устанавливаем драйверы Nvidia

Оборудование Nvidia – популярный выбор среди пользователей Linux, поскольку оно предоставляет лучшую 3D-производительность, которую только можно купить. Если вы любите играть или вам нужно работать с ПО для 3D-моделирования, вопрос производительности является существенным. Необходимое условие для инсталляции драйверов Nvidia – установленный исходный код вашего Linux-ядра. Прежде чем приступить к установке драйвера, откройте менеджер пакетов вашего дистрибутива и установите пакет исходных текстов. Есть шанс обойтись и без этого, если Nvidia уже предоставляет двоичный драйвер для вашего ядра. В противном случае просто следуйте этим шагам, войдя в систему как root.



» Скачивание драйвера

Закройте *X Window* или загрузитесь в консоль. Вы всегда сможете скачать последний драйвер из консоли, используя web-браузер *Lynx*. Введите lynx **http://www.nvidia.com/object/unix.html**, найдите ссылку для скачивания и нажмите <d>.



» Запуск скрипта установки

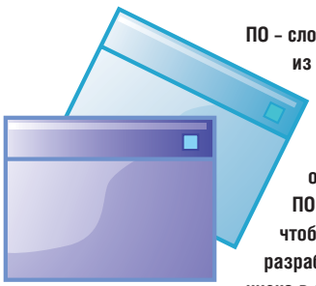
Введите **sh**, а затем – имя полученного файла. Если инсталлятор возмущается, что вы не закрыли *X*, удалите **/tmp/.X11-unix** и **/tmp/.X11-lock** и попробуйте снова. Процесс установки выполняется автоматически; инсталлятор изменит ваш файл **Xorg.conf**.

» Проверка

Модуль ядра будет собран и установлен. Для гарантированной его загрузки вы можете ввести **modprobe nvidia**. После этого остается либо перезагрузить машину, либо запустить менеджер входа в систему – *gdm* или *kdm*.

Программное обеспечение

Тот факт, что ПО в Linux свободно, позволяет сделать многое. Но свободное ПО зачастую идет без свободной поддержки...



ПО – сложная штука. Невозможно охватить все, поскольку Linux – одна из самых распространенных и используемых ОС на планете. И она бесконечно разнообразна. Каждый из нас так или иначе сталкивается с программными проблемами. И все-таки многие проблемы возникают чаще, чем должны бы. В основном это связано с обычным для свободного и открытого ПО недостатком документации – все меняется слишком быстро, чтобы технические писатели поспевали за разработкой. А если вы разработчик-одиночка, документация обычно расположена слишком низко в списке приоритетов.

Хороший пример – композитный менеджер *Compiz* для сногсшибательных 3D-эффектов. Год назад он был на этапе формирования и вышел в январе 2006 года. Но за последние 12 месяцев программа созрела, к ней появились расширения, и в доказательство реального прогресса, проект распался на два (вторая ветвь известна как *Beryl*). Однако, несмотря на весь этот «прогресс», от жаждущих трехмерного куба чаще всего приходится слышать вопрос: а как заставить его работать? А жаждут практически все.

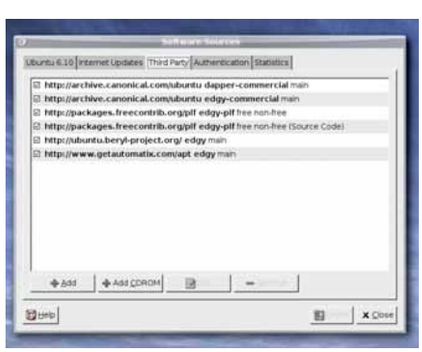
«Ломаем» Compiz

Для тех, кто хочет получить последний *Compiz*/*Beryl*, жизнь недавно стала заметно проще, благодаря тому, что большинство современных дистрибутивов использует сервер *X.org* с активированным *AIGLX*. Использование *AIGLX* означает, что больше не нужно менять используемый по умолчанию сервер *X Window* на *Xgl*. Первоначально для многих пользователей это было основным камнем преткновения. Теперь остается лишь убедиться, что для вашей видеокарты есть драйверы, поддерживающие аппаратное ускорение (*см. раздел «Оборудование»*), и один из пакетов – *Compiz* или *Beryl* – установлен с помощью менеджера пакетов вашего дистрибутива.

Если пакеты не предоставляются по умолчанию, потребуется добавить сторонний репозиторий. Например, в последней версии Ubuntu нужно добавить репозиторий Beryl (Ubuntu Beryl Repository, **http://ubuntu.beryl-project.org**) для загрузки пакетов ПО, прежде чем приступать к установке. Затем выполните **Beryl-manager** из командной строки – и текущая оконная рамка превратится в трясущееся желе.

Идем в ногу со временем

С «красотой» мы разобрались; следующий наиболее популярный вопрос – как устанавливать файлы из Интернета. Ответить на него сложно, поскольку здесь нет простого решения. В Linux нет эквивалента скачивания программы **setup.exe** и ее запуска, как делается в Microsoft Windows. Менеджер пакетов дистрибутива ближе всего подходит к установке «в один клик», но он полностью зависит от дистрибутива, обеспечивающего пакеты для нужных вам приложений. Mandriva пытается отслеживать запросы публики и предоставлять пакеты для наиболее востребованных приложений, а Ubuntu и Debian включают тысячи прекомпилированных пакетов для установки.



› Ubuntu позволяет добавлять различные источники для вашего ПО через приложение *Software Sources*.

Но с появлением нового релиза вы застреваете. Каждому охота скачать и установить последнюю версию приложения, но если вам нужны официальные пакеты, то придется подождать следующего официального релиза выбранного вами дистрибутива. Это одна из причин, почему так много людей мечутся от одного дистрибутива к другому – они бегут с корабля, чтобы получить новейшую версию пакета, недоступную в настоящее время в их текущем дистрибутиве.

Несмотря на это, есть несколько способов получить последнюю версию желанного ПО. Например, *Mozilla Firefox* доступен в виде статически собранного исполняемого файла – вы можете запустить эту программу сразу же, как скачаете, без размышлений о зависимостях и сценариях настройки. Вы также обнаружите, что готовые пакеты часто предоставляют сторонние разработчики (например, команда *Beryl* – свое композитное ПО), и установка их разработок обычно сводится к добавлению нового сервера в вашем менеджере пакетов до того, как вы решите установить свежее ПО.

С исходным кодом на «ты»

Если вам так уж нужна наисвежайшая версия, временами нет другого способа ее получить, кроме как скачать «голый» исходный код. Вид его зависит от вкуса автора, но наиболее общий метод распространения программ – упаковка исходного кода в файлы **.tar.gz** или **.tar.bz2**. Собирая свое ПО, легко нарваться на неприятности, и вообще-то это следует рассматривать как крайность – но большинство приложений можно собрать, следуя простой стандартной процедуре.

Первый шаг – убедиться, что на вашей машине установлено необходимое для сборки окружение: не только инструменты компилятора GNU, но и версии «для разработчика» всех библиотек, нужных вашей программе. Это зависимости в чистом виде, и вы можете определить их, читая сопроводительную документацию. Поняв, какие пакеты вам нужно установить, убедитесь, что вы устанавливаете 'dev'-версию из

«Скорая помощь» по программам

» Ближайший аналог инсталляционных файлов **setup.exe** в Linux – это Autopackage. Их довольно сложно делать, но если они доступны, то ПО устанавливается просто вихрем. Autopackage-файлы для последних релизов *Gimp*, *Gaim* и *Inkskape* находятся на **http://autopackage.org**.

» Если вам нужно запускать системный скрипт во время загрузки, просто разместите ссылку на этот скрипт в каталоге **/etc/rc.local**. Как вариант, пользователь может добавить скрипт к своей Gnome-сессии из меню System > Preferences > Sessions, или кинуть ссылку в **~/.kde/Autostart** в случае KDE.

» Настройки рабочего стола можно найти в директориях **.gnome** или **.kde** в вашем домашнем каталоге. Копирование всего содержимого этих каталогов в другую систему перенесет большинство ваших системных установок, при условии, что вы используете одинаковые версии рабочих столов [и не забудете сменить права доступа к файлам: пользователь «user» в дистрибутиве А и дистрибутиве Б – разный пользователь, – прим. ред.].

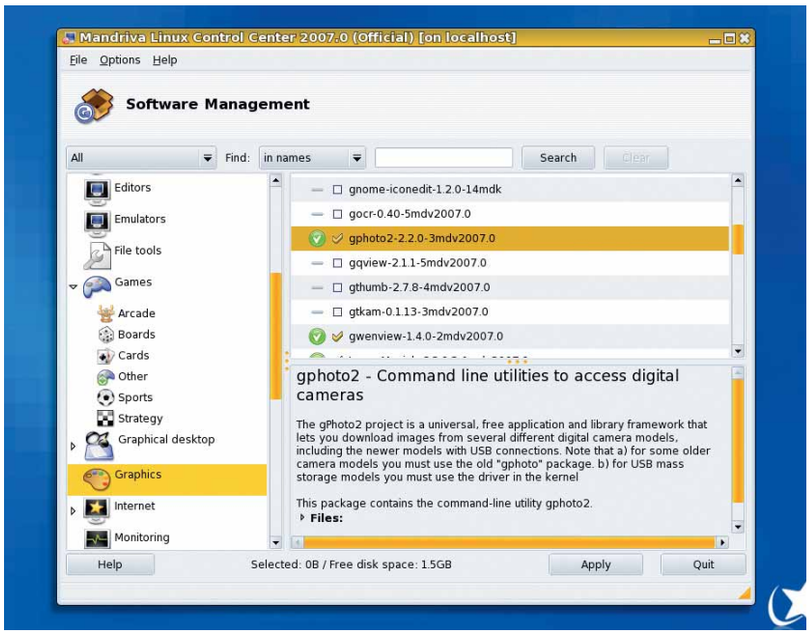
» Менеджер пакетов можно использовать как для установки, так и для удаления ПО. Если вы ощутили недостаток места на диске, вспомните, что можно освободить немного места, удалив неиспользуемые приложения.

» Вы можете конвертировать RPM в Deb и наоборот, используя удобную утилиту под названием *Alien*. Создаваемые ею пакеты в критических для системы приложениях использовать не следует, но это может реально помочь, если никакие другие пакеты не доступны. См. **http://kitenet.net/~joey/code/alien.html**.

менеджера пакетов вашего дистрибутива – это те самые файлы «для разработчика», которые нужны для сборки вашего приложения.

При установленных зависимостях большинство приложений требуют запуска **./configure** и **make** в созданном каталоге. Именно здесь и встречаются проблемы, когда процесс сборки не может найти библиотеку или файл, от которых он может зависеть. Хитрость тут в том, чтобы найти в выводе место возникновения первой ошибки и внимательно прочитать текст, чтобы найти, какой файл пропущен.

Типичная ошибка может выглядеть как-то так: **'libguile.h**: No such file or directory'. В данном случае не найден файл libguile.h. Как правило, вы сможете определить, какой пакет нужно установить, поискав этот файл в менеджере пакетов вашего дистрибутива или даже в Интернете. В случае с Debian/Ubuntu, хороший ресурс – www.debian.org/distrib/packages. Чтобы найти необходимый для решения проблемы пакет, вы можете ввести 'libguile.h' в поле «Search The Contents Of Packages» на этой странице. В нашем примере нехватает пакета **libguile-dev**, его можно установить с помощью *Synaptic*. С другой стороны, владельцы Mandriva могут поискать файлы, используя графический менеджер пакетов или набрав в командной строке **urpmf libguile-dev** от имени root. Компиляция проблемного ПО – это вопрос внимательного просмотра сообщений об ошибках, пока не будут выполнены все недокументированные требования.



Пестрота мультимедиа

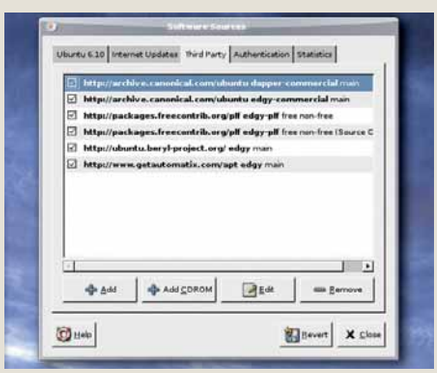
Что касается установки ПО, большинство пользователей, заполучив свой сияющий новизной Linux, первым делом желает либо послушать музыку в формате MP3, либо посмотреть в онлайне видео, закодированное с использованием проприетарного формата. Проблема заключается в том, что команды большинства дистрибутивов не хотят вступать в конфликт с законом, включая технологии, которые могут чем-то нарушить права на интеллектуальную собственность. MP3 – хороший пример, поскольку патент на его дизайн активно навязывается французской технологической компанией Thomson Consumer Electronics. Многие команды выбирают безопасную тропинку, оставляя его за пределами своего дистрибутива, поэтому-то открытые форматы, типа Ogg Vorbis, столь хорошо поддерживаются поставщиками дистрибутивов.

Несмотря на эти ограничения, обычно есть возможность заставить большинство файловых форматов прекрасно работать в Linux. Например, разработанная сообществом версия *RealPlayer* для Linux включает лицензированное воспроизведение MP3 (см. www.real.com/linux). Но есть также возможность добавить поддержку различных форматов при помощи пакетов и репозиториев сторонних производителей. Активация Multiverse в Ubuntu позволит вам скачать пакет кодеков Win32, включающий поддержку всевозможных проприетарных форматов, а Penguin Liberation Front поддерживает также репозитории для Mandriva.

› Новый инструмент пакетного менеджмента в Mandriva – один из простейших в использовании, он делает обработку зависимостей почти безболезненной.

Шаг за шагом: Добавление репозитория в Ubuntu

Один из наиболее частых вопросов, получаемых в Linux Format – как активировать сторонние или неподдерживаемые репозитории для вашего дистрибутива. Понятно, что процесс в каждом дистрибутиве свой, но поскольку Ubuntu – это хороший выбор для начинающих, то вот вам инструкция.

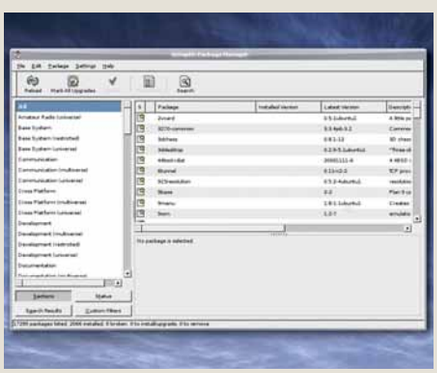


» Выбор источников

Запустите менеджер пакетов *Synaptic*, затем отметьте Settings > Repository. Откроется новое приложение Select Sources, в котором вы сможете активировать новые репозитории на вкладке Third Party или добавить собственный, щелкнув на New.

» Скачивание

Когда бы вы ни сделали изменения в базе пакетов, *Synaptic* потребует обновить список доступных пакетов. Просто щелкните на Reload в *Synaptic*, чтобы скачать новый список пакетов с различных репозиториев.



» Поиск!

Каждый поиск приложения теперь будет охватывать новые источники пакетов, используете ли вы *Synaptic* или apt в командной строке. Это самый безопасный способ добавить ПО, но сторонние пакеты могут оказаться проблемными.

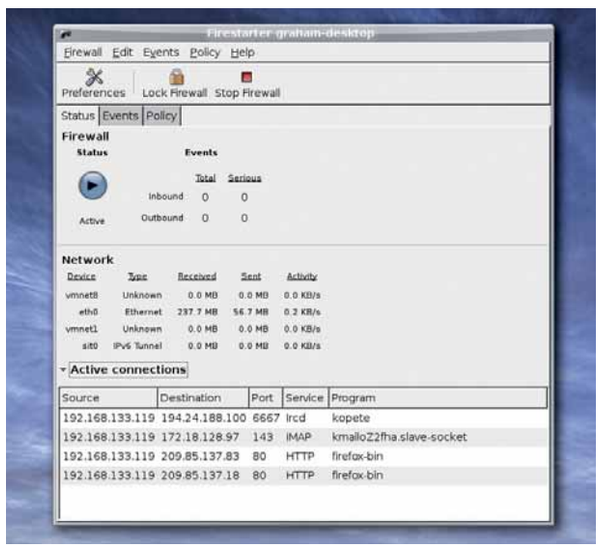
Сеть и администрирование

В наши дни грош цена тем, кто не подключен к Интернету.

В Linux вы недалеко уйдете без подключения к сети. Несмотря на это, сейчас нет стандартного приложения для его настройки. В каждом дистрибутиве свой подход к поддержке сети, и как правило, предлагается собственная панель конфигурации или скрипт для настройки вашего соединения, а это вам не сильно поможет, если что-то поломается.

Если ваша машина отказывается выходить в Интернет, можно попробовать несколько вещей. Наиболее обычная помеха в плохо настроенной сети – ваш компьютер не может преобразовать имя сервера в IP-адрес, необходимый для установления соединения. То есть если вы наберете что-то типа ‘bbc.co.uk’ в своем браузере, сетевой уровень, отвечающий за установление соединения, будет нуждаться в конвертации его в номер, или IP-адрес. Например, bbc.co.uk на самом деле соответствует IP-адресу 212.58.228.155.

Одно в другое конвертирует DNS, сервер доменных имен. Если он не настроен, вы не сможете просматривать страницы, используя доменные имена, но тем не менее сможете использовать IP-адреса. Вот и первый ключ к решению проблемы – если доступ по IP-адресам работает, а по доменному имени – нет, значит, проблема в настройке DNS. Обычно это означает, что вы ввели неправильный адрес сервера. Адрес DNS вашего провайдера размещается в файле **/etc/resolv.conf**, и вам нужно проверить, что после слова ‘nameserver’ указан правильный адрес.



› Использование инструмента типа *Firestarter* для настройки брандмауэра сильно упрощает установку общего интернет-соединения.

Общее соединение

У многих из нас сейчас больше одного компьютера, которые без интернет-соединения полезны лишь наполовину. Вы можете использовать беспроводную сеть, но есть также возможность разделять соединение, используя Ethernet-кабель, если два компьютера достаточно близко друг от друга. Если у вас нет коммутатора между этими двумя компьютерами, важно, чтобы используемый вами кабель был «перекрестным» (crossover), а не стандартным.

В некоторых дистрибутивах общее соединение создается открытием панели управления на машине, подсоединенной к Интернету, и активацией соответствующей опции. Например, в Mandriva общее соединение может быть активировано на страницу Network & Internet в центре управления. Но в других дистрибутивах, особенно в Ubuntu, вам придется выполнять настройку вручную.

Намного проще настроить обе машины, если сетевые соединения используют статические адреса, а не назначаемые автоматически – особенно для машины, с которой вы хотите разделять соединение. Потому что если вы хотите, чтобы машины использовали общее соединение, имея автоматически назначаемый IP-адрес, то придется учитывать дополнительную сложность в лице DHCP-сервера. Общее интернет-соединение складывается из двух элементов: маршрутизации пакетов (forwarding) и трансляции сетевых адресов (NAT), оба они реализуются с помощью *Iptables* [строго говоря, это не совсем верно. Часть ядра, отвечающая за маршрутизацию, может работать и без *Iptables*, – прим. ред.]. Это сетевой инструмент, работающий на уровне ядра и отвечающий за управление сетевыми пакетами – обычно известный как брандмауэр. А установка брандмауэра – самый простой способ настроить общее интернет-соединение.

Firestarter – одна из наиболее популярных утилит настройки брандмауэра для Linux. При первом старте «мастер» быстрого запуска проведет вас по настройкам брандмауэра и созданию общего интернет-соединения. Он берет на себя большую часть забот по настройке *Iptables*, а также предоставляет в реальном времени информацию о том, какой именно процесс работает с Интернетом.

Потерянный пароль

Пароли не особо важны, если ваш компьютер стоит себе в углу, без подключения к интернету, и вы его единственный пользователь. Но как только вашу машину захочет использовать кто-то еще или вы пожелаете войти в

«Скорая помощь» по сетям и администрированию

» Существует множество беспроводных устройств, не поддерживаемых в Linux. Но часто их можно заставить работать, используя комбинацию Windows-драйвера и превосходное Linux-приложение по имени *Ndiswrapper*. Он объединяет эмуляцию Windows с магией Linux, чтобы вернуть оборудование к жизни.

» Брандмауэры зачастую понимаются неправильно, потому что пользователям Windows вдалбливают, что он должен быть у них установлен. Истина же в том, что если у вас нет портов или сетевых сервисов, запущенных на вашей машине, то нужды в брандмауэре никакой нет – вот почему Ubuntu не поставляется с преднастроенным брандмауэром.

» Если вам нужно администрировать вашу систему удаленно, попробуйте установить программу *Webmin*. Она позволит настраивать почти все аспекты работы вашей машины, от web-сервера до пользовательских учетных записей, и все это через браузер на удаленной машине.

» Сетевые USB-устройства, от старых телефонных модемов до новых ADSL, создают проблемы. Вы можете заставить их работать, но обычно это не стоит затраченных усилий. Гораздо проще купить дешевый ADSL-маршрутизатор и использовать подключение по Ethernet-кабелю – и никакой специальной настройки не потребуется.

» Таблицу маршрутизации (слегка запутанную) можно просмотреть, набрав **route** от имени root. Если у вас больше одного сетевого устройства, убедитесь, что только один маршрут отмечен как используемый по умолчанию. Можете удалить любые дополнительные маршруты по умолчанию, набрав **route del default eth0**, только замените ‘eth0’ на имя соответствующего устройства.

Лучшие советы из Башен LXF

Говорит Майк

«Одна из наиболее частых проблем, о которых я слышал – запись CD. Из-за ошибки в ядре Linux некоторые CD (особенно загрузочные диски дистрибутивов) записываются под Linux неправильно, пока вы не воспользуетесь особыми опциями. В частности, эта проблема имеется с дисками Fedora Core.

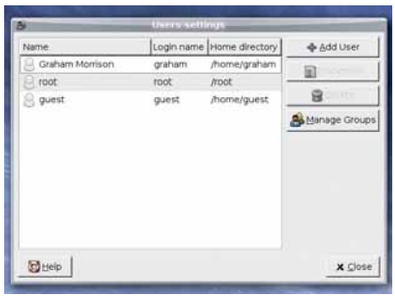
Когда записывается CD с дистрибутивом Linux, используйте опцию 'padding' в своей программе записи CD, при этом оставляется небольшое пустое пространство на диске. Например, если вы используете инструмент командной строки *CDRecord*, вам нужно определить опцию '-pad padsize=63s', что даст гарантию, что этот диск будет загружаться, несмотря на ошибку ядра. В *K3b* вы также можете установить опцию 'padding'.»

Говорит Пол

«Работать с Windows-пользователями не слишком легко, но чтобы разделять с ними файлы, особых усилий не требуется. Простейший путь прочитать Windows-папку – это использовать опцию Gnome Connect To Server (ищите ее в меню Places). Когда появится окно, выберите Windows Share как тип сервиса, затем заполните IP-адрес сервера и имя общей папки (обычно это просто название каталога), и готово!»

Говорит Ник

«Google – превосходный инструмент для поиска ответов: есть все шансы, что кто-то уже сталкивался с подобной проблемой до вас. Но чтобы не тратить уйму времени, разгребая не относящиеся к делу соответствия, будьте точны и заключайте искомые данные в двойные кавычки: "Error 17: Cannot find device" гораздо вероятнее поднимет вверх нужный результат с кавычками, чем без них. Помимо этого, лучший способ утрясти свои проблемы – это понять, что же идет неправильно: dmesg, содержимое **/var/log** и форумы *Linux Format* должны стать вашими друзьями.»



› Даже если вы присматриваете за системой, где больше одного пользователя, сейчас гораздо проще использовать графический инструмент администрирования, а не консоль.

сеть, хороший пароль становится жизненно необходимым для безопасности вашей системы. Хороший пароль, однако, плох тем, что его сложно запомнить. Самый популярный пароль, вероятно, 'password1', который как минимум учел желательность смешения букв и цифр. Если вы еще помните свой пароль root, достаточно войти в систему как root и назначить пароль пользователя командой passwd username. А вдруг пароль для учетной записи root вы тоже забыли? В случае Ubuntu, где и учетной записи root нету, забыть свой пользовательский пароль означает здорово влипнуть.

Тогда придется перезагрузить вашу систему либо в безопасный режим, либо в режим восстановления. Обычно это применяется для решения проблем с загрузкой, но также полезно и для получения доступа с правами root без пароля. Если вы пользователь Debian или Ubuntu, при виде загрузочного меню *Grub* нажмите <Escape> и выберите загрузку ядра в режиме восстановления. По завершении процесса загрузки можно будет изменить пользовательский пароль обычным образом.

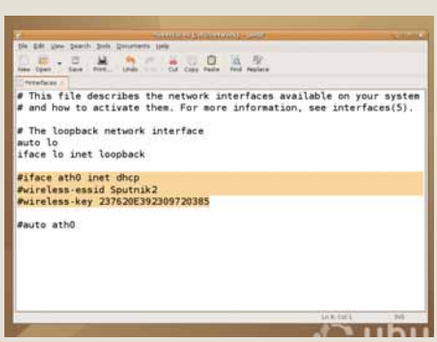
...теперь отдыхайте!

Интернет полнится людьми, перенастраивающими свои дистрибутивы Linux, и мы отметили в начале, что для некоторых это весьма привлекательно. Продравшись сквозь проблемы и сумев самостоятельно их устранить, вы испытаете чувство подлинного удовлетворения. Это обряд посвящения для каждого пользователя Linux.

Мы надеемся, что вы насобирали достаточно информации, способной помочь вам при любых проблемах; мы, как минимум, дали вам хорошую отправную точку для поиска решения. Но мы будем рады, если вы обратитесь к нам со своими сложностями. Можете также посетить наши дружелюбные форумы (www.linuxforum.ru). Linux зиждется на чувстве взаимопомощи; именно так мы делаем мир лучше. LXF

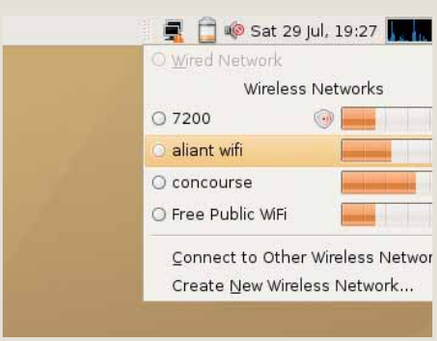
Шаг за шагом: Использование менеджера настройки

Долгое время беспроводные сети причиняли в Linux настоящие мучения. К счастью, две утилиты берут на себя большую часть головной боли, связанной с этим процессом. Менеджеры настройки сети Gnome и KDE – это нововведения, которые, по сути, идентичны. Они выступают как интерфейс между вашим сетевым оборудованием и беспроводной сетью и позволяют вам сканировать доступные сети, сохраняя профиль для тех, которые вы будете использовать. Это значительное усовершенствование по сравнению с панелью управления старых дистрибутивов. Для начала воспользуйтесь своим менеджером пакетов, чтобы установить либо Gnome network-manager, либо KNetworkManager.



» Подготовка

Вам может потребоваться отредактировать файл **/etc/network/interfaces**. Он содержит вашу текущую фиксированную беспроводную конфигурацию, которую нужно будет закомментировать, чтобы менеджер настройки сети смог использовать собственные установки.



» Запуск

Установленный и запущенный менеджер настройки сети можно будет найти на панели рабочего стола. Щелкните по иконке, чтобы открыть список доступных сетей, наряду с индикатором уровня сигнала и значком замка или щита, отмечающим зашифрованные сети.

» Соединение

Если вы выбираете незашифрованную сеть, интернет-соединение установится автоматически. Если сеть требует ввода ключа, появится окно запроса, чтобы вы ввели в него дополнительную информацию. Если данные введены правильно, соединение будет установлено.

За кулисами Perl 6

Perl – один из самых популярных существующих языков программирования. Одна из причин этой популярности – продуктивные отношения между создателем языка Ларри Уоллом [Larry Wall] и его главным сообщником **Дэмьеном Конвеем** [Damian Conway].

В толпе разработчиков Дэмьена вы узнаете сразу. Это – парень в серых брюках, черной футболке и с толпой вцепившихся в него программистов Perl. Если Ларри Уолл – Янь, то он – Инь, адвокат дьявола. Ларри и Дэмьен вместе трудились в кузнице, где был выкован Клинок Perl и где его превратили в Perl 6; от него зависит будущее всей компьютерной братии. **Грэм Моррисон** получил задание встретиться с Дэмьеном и уточнить, что происходит с Perl 6.

Linux Format: На что сейчас в основном уходит ваше время?

Дэмьен Конвей: Примерно 50% занято банальным зарабатыванием на жизнь. То есть я готовлю занятия и презентации, а затем – путешествую по всему миру, доставляя их в нужные места. Еще 20%, наверное, уходят на поиски работы: контрактов и людей, которые хотят, чтобы я выполнил для них некое дело – то есть, логистику, необходимую, чтобы все получилось.

А остальное время почти полностью посвящено разработке Perl 6. На данный момент – это совместная работа с Ларри над финальными штрихами, чтобы убедиться в том, что все части осмысленны и хорошо работают вместе, и я помогаю ему сложить оставшиеся элементы головоломки.

Довольно интересно, что эта завершающая стадия работы над Perl 6 немного сложнее, чем стадия общей разработки, потому что теперь почти все уже на своих местах, и большинством сделанного мы очень довольны, есть несколько мелочей, которые должны быть не хуже всего прочего, а у нас меньше возможностей для их разработки. Так что именно поиск нужного решения для этих последних штрихов кажется мне таким интересным, это своего рода проба сил.

LXF: А преимущества у такого способа работы есть?

ДК: Это своего рода вызов, но да, я думаю, преимущества есть. И преимущество состоит в том, что мы не связаны никакими условиями. Если бы мы работали в какой-то фирме, у нее были бы свои цели, связанные с Perl 6, и нам, как сотруд- »

Фото: Джейсон Каплан [Jason Kaplan].

О ТОМ, КОГДА ОСТАНОВИТЬСЯ

«Наступает такой момент, когда можно сказать: Да, теперь все срослось.»

» никам фирмы, пришлось бы в первую очередь заботиться об этих целях. А мы работаем на сообщество, и можем сконцентрироваться на целях Perl 6, важных для сообщества, и эти цели куда более многообразны, чем могли бы быть цели любой, самой крупной фирмы.

Люди говорят: «Мы разрабатывали Perl 6 шесть лет. Почему же он еще не появился?» А я думаю, что фактически, это преимущество – у нас было время разработать то, что будет самым лучшим на следующие 20 лет. Конечно, с точки зрения PR это не слишком хорошо, но с точки зрения языка, который мы создаем (на самом деле, уже создали), это настолько хорошо, что лучше и сделать нельзя. Подумайте, какого калибра люди участвуют в проекте; сразу станет понятно, что все это – просто здорово.

Есть фраза, которую я повторяю чаще всех остальных: «Вот как это в Perl 6». Я вижу, какое волнение написано на лицах людей, если они в течение получаса видят то, что мы внесли в язык, честно, я вижу на их лицах вожделение: «Почему его еще нет? Почему нельзя пойти домой и приняться за него прямо сейчас?» …По-моему, самое прекрасное, что произошло за два года с нашей прошлой беседы [Интервью, LXF62] – это что у нас был проект Pugs с Одри Танг [Audrey Tang], которая проложила нам дорогу, и сейчас вы можете пойти и сделать это [использовать Perl 6].

Когда я делаю презентацию, последний мой слайд – это всегда URL кода Pugs. Люди расходятся и загружают его, и где-то через полчаса они уже могут поиграть с настоящим кодом Perl 6. Все это вызывает большой ажиотаж и энтузиазм. Да и много огорчений тоже, потому что людям нравится этим заниматься, но ведь это не готово для выпуска и не обладает той же скоростью, что Perl 5.

Но в общем и целом меня крайне радует достигнутое, и меня крайне радует, что у нас было достаточно времени сделать работу хорошо. И во многом из-за того, что на нас не давил корпоративный график работ.

LXF: Повлияло ли на процесс разработки за последние шесть лет появление PHP или Ruby?

ДК: Я не совсем уверен, что эти два оказали какое-то влияние именно в пределах указанного интервала. Конечно, Ruby сильно повлиял на Perl 6; я бы ничего не стал убирать из того, что Мац [Юкихиро Мацумото, Yukihiro Matsumoto, автор Ruby] сотворил с Ruby. Многое из того, что он позаимствовал в Perl и запихнул в Ruby, снова вернулось в Perl. И я уверен, что если бы не было Ruby, Perl не смог бы стать настолько объектно-ориентированным, как Perl 6.

Если быть предельно честным, то большинство языков, появившихся в течение упомянутого периода, не особо отличались от ранее существовавших. Я думаю, что наше мышление изменилось из-за усиления некоторых парадигм и некого понимания, пришедшего к нам в процессе программирования – что работает, а что не работает. По примеру на то и на другое: за этот период вроде как расцвело аспектно-ориентированное программирование. Люди осознали, что хотя объектно-ориентированному программированию есть что предложить, язык не может быть полным, если в нем нет механизма, ортогонального к структуре классов, который позволял бы добавлять необходимые функции, но не принадлежал к иерархии. И что здорово – мы смогли взять это и внедрить. Мы сказали: «Ладно, Perl нужна поддержка аспектно-ориентированного программирования, поэтому встроим такой механизм прямо в ядро языка…»

LXF: Так вот куда попадает Pugs?

ДК: Не думаю. Я думаю, что сюда попадает компонент под названием PGE, это грамматический движок. И он действительно хорошо подходит, потому что мы разработали грамматику Perl 6 таким образом, что вы получаете всю мощность синтаксического анализа самой лучшей породы, какую только можно найти. Раньше все грамматические конструкции, такие как *Lex* и *Yacc* на раннем этапе или более современные вещи, вроде *Antlr* – они всегда были отделены от языка. Они всегда были инструментом, который применялся в языке, или применялся для создания языка.

Мы знали, что у Perl лучшие регулярные выражения на свете, но до следующего уровня он не поднимался: до грамматического уровня, на котором можно, используя все это, легко создать интерпретаторы языков. Вот мы и внесли это в язык.

Такая вещь – рост нашего понимания того, что нужно для хорошей разработки и развитие различных парадигм языковых функций (хотя большинство из них имеются в большинстве натуральных языков мира) – позволяет нам, на основании этой половины десятилетия, заявить: «Мы считаем, что это будет хорошо, и это будут важно и через 20 лет, поэтому Perl 6 мало того, что сможет просто выполнять все эти вещи – он должен их поддерживать».

Наши мысли о разработке языка во многом получали информацию от этого. Язык должен быть легок для тестирования, для синтаксического анализа, для рефакторинга. Да, тестировать программы нужно; но не просто проверять правильность функционирования, а убеждаться, что тест охватывает все возможности. Поэтому нужен язык, в котором было бы видно внутреннее представление всех ветвей в программе, а для этого нужно, чтобы внутреннее представление программы было в самой программе.

О СУПЕРФУНКЦИИ PERL 6

«Вы можете использовать объектно-ориентированное программирование так же легко, как в Java и C++.»

LXF: Как вы узнаете, что пора ставить точку?

ДК: Какой чудесный, чудесный вопрос. [Долгая пауза] Знаете, меня никогда об этом не спрашивали, а это ключевой вопрос. Как узнать, что всего уже достаточно, чтобы не переборщить?

Все эти годы мы придерживались принципа: никогда не включать функций, которые служили бы только для одной цели. Если нечто претендовало на включение в Perl, значит, должно было годиться для решения пары задач. Оно должно было улучшать то, что уже было в языке, плюс обеспечивать дополнительную функциональность, то есть давать еще и чистую прибыль. Потому что нужно было считаться с той самой проблемой, о которой вы сказали, а именно: «Уймись, не то язык чересчур разбухнет».

Но по большому счету, вы правы: есть этот неиссякаемый фонтан новых технологий в мире программирования, как в сфере теории, так и в сфере практических технологий разработки. Так как же решить, что пора остановиться? Если честно, я думаю, ответ такой: когда устанешь! Нет, я думаю, когда достаточно глубоко изучил создание языков, то тебе совершенно ясен момент, когда можно сказать: «Да, теперь все срослось, теперь это – одно непрерывное полотно». И когда это происходит, надо остановиться. Потому что если зайти слишком далеко, то начнешь включать в язык какие-то лишние штуки, которые там совершенно на данной стадии не нужны.

Вот вам ответ: мы останавливаемся, когда появляется ощущение, что все срослось и мы сделали все, что хотели сделать. Естественно, мы не касались новшеств, появившихся месяца три назад, но зато мы сделали другое – по нашему мнению, мы создали новую версию Perl настолько расширяемой, что новшества, которые появляются сейчас, или появятся через два года, через пять лет, очень легко будут интегрированы в Perl 6, так что, возможно, Perl 7 уже не понадобится.

LXF: А бывают разногласия между разработчиками, о том, что важно и когда надо остановиться?

ДК: Ну что вы, у нас полное согласие всегда и во всём. [Смеется] Да нет, по-моему, вполне здоровая ситуация, когда спорят на тему «как лучше это сделать?» или

«подходит это сюда или нет?». Если вы посмотрите на списки рассылки, у нас там очень много талантливых людей, они проверяют наши решения и рассматривают их под разным углом, так вот, если на эти списки взглянуть, вы увидите массу разногласий по теме, как правильно работать.

Так что, я думаю, разногласия – это очень ценная вещь. А в конечном итоге нам помогает тот факт, что Perl – это то, что о нем говорит Ларри.

LXF: Это здорово.

ДК: Ну, должен быть кто-то, в кого все упирается, иначе никуда не придешь. И почему в качестве такого человека хорошо иметь Ларри – Ларри начисто лишен эго, применительно к данной разработке. Он хочет слышать достойные идеи от других, и я видел, как он целиком соглашался с предложениями людей, которые вообще впервые оказались в списке [рассылки]. Ему не важно, кто выдвинул идею, ему важно только, насколько эта идея разумна.

LXF: Вы считаете, что после Perl 6, разработка Perl станет в большей степени ориентирована на сообщество?

ДК: Думаю, это весьма вероятно. Именно это мы наблюдаем в Perl 5. В конце концов, основная разработка Perl 5 сейчас ведется людьми из Perl 5 porters. И хотя Лари по-прежнему обладает правом полного вето, он крайне редко им пользуется. И он обычно оказывает всяческую поддержку, если они хотят что-то сделать, потому что Perl 5 porters отлично знают код.

LXF: А какие части Perl 6 вы любите больше всего?

ДК: С предвзятой точки зрения, один из моих любимых разделов – это функция объединений (junctions). Потому что я часто думаю о том, что объединения – мой единственный вклад в компьютерные науки. Объединения – это такая разновидность принципа квантовой суперпозиции в Perl 6: например, бывают величины, равные 5, 6, 7 и 8 одновременно. Это необыкновенно удобно, потому что если у вас есть длинный список всяких вещей, вы можете свести их в одну величину, а затем достаточно одного теста для проверки на равенство любой из них.

> ОБ ОТКРЫТОМ КОДЕ:
> **«Проблемой будет — как конкурировать, не становясь конкурентами.»**

Например, у вас есть список неприемлемых паролей, и вы можете создать одну скалярную величину, которая равна любому из них. Затем вы можете сказать: «Дайте мне ваш пароль, и если он равен этой одной скалярной величине, значит, он равен и любому [из неприемлемых паролей]». Дело в том, что благодаря такой структуре можно – по крайней мере, теоретически – параллелизовать этот тест. Тут известно, что вам надо просто проверить равенство с любой величиной, а они совершенно независимы, и если это не так – применительно к любой из них – у вас есть возможность вернуться к исходной точке кратчайшим путем. Это же столбовая дорога к параллелизации. Вы можете просто ввести в Perl: **if $password eq any(@existing_passwords)**. И вы получаете возможность параллелизовать это, причем написать это на практически обычном английском языке.

Я горжусь, что эта идея принадлежит мне, и я был в восторге от того, как её приняли и усовершенствовали другие члены команды, прежде чем ввести в язык.

А что до пристрастий, я люблю всякие милые мелочи. Например, то, что теперь у нас есть функция «print line» в Perl, потому что в 70% всех напечатанных выражений приходится в конце ставить новую строку. Меня это просто бесило! А теперь есть маленькая функция say – вы просто пишете **say $whatever**, а она сама вставляет перевод строки. Штука тривиальная, такое можно ввести в любой язык, но немного найдется языков, которые это сделали, да ещё столь изящно.

LXF: Вы довольны, что Perl 6 усилил объектную ориентированность?

ДК: Да, думаю, было бы неправильно не признать этого. В смысле практического применения языка, факт наличия формализованной объектно-ориентированной структуры классов означает, что можно заниматься ОО-программированием; это легко делать в Java, C# и C++, а теперь не менее легко и в Perl, и так же понятно… На самом деле, мы внесли в него такое, чего нет во всех этих языках. Я думаю, это будет очень важным отличием. Видимо, я меньше думаю об этом, чем мог бы, потому что почти не занимаюсь объектно-ориентированным программированием. Но я знаю, что для тех, кто им занимается, это будет, вероятно, супер-функция в Perl 6. То, что теперь можно писать классы с должным образом объявленными компонентами атрибутов и методами, наверное, окажет самое сильное позитивное воздействие из всего, о чем мы говорили. Поэтому, конечно, меня это очень радует, но я об этом даже не думаю, потому что в Perl 6 это вообще-то фундаментально.

За Дэмьеном охотятся ради его презентаций: на фото – его выступление *Da Vinci Codebase* на OSCon 2006.

LXF: Как вы думаете, с какими проблемами столкнется сообщество Open Source в следующие пять лет?

ДК: [Долгая пауза] Думаю, проблема для сообщества Open Source в следующую половинку десятилетия – как конкурировать, не становясь конкурентами. Мы должны сохранить веру в то, что философская основа нашего существования, того, чем мы занимаемся, действительно правильна, правильна настолько, что мы сумеем выжить, продолжая делать то, что делаем. Нам придется решать, как стать мейнстримом и при это не угодить ни в одну из ловушек мейнстрима – и не поддаться существующей практике. Меня не удивляют фирмы, которые закрывают код, вместо того, чтоб открыться настолько, насколько возможно; по-моему, у них просто сдают нервы. Думаю, вполне понятно, что множество начинаний уже продемонстрировало: с философией Open Source, оставляя код открытым, вполне возможно жить, да ещё и получать прибыль.

Сейчас Open Source становится мейнстримом, в огромной степени благодаря таким программам, как *Firefox*. Linux… наверное, не до такой степени, как *Firefox*, потому что *Firefox* – это интерфейс, через который обычный человек, не компьютерный эксперт, ежедневно сталкивается с разными хитрыми штуками, и все больше людей этим занимается. Я хочу сказать, заметно, что доля Linux на рынке тоже растет, но, наверно, не такими темпами. Ведь в *Firefox* можно работать и под Windows, наверное, причина в этом. LXF

Узнать больше!

«Perl – это детище Ларри, а я вроде его злобного братца, стало быть, я и дядюшка...» Подробности – на www.linuxformat.co.uk/mag/conway.

Перышки вам в шляпу

Помните: изменения в файле конфигурации Apache не вступят в силу, пока вы не перезапустите сервер!

Решили научить свой сервер *Apache* новым трюкам? Узнайте у **Пола Хадсона** про три его любимых модуля *Apache* для web-суперобслуживания.

Приспособить Apache для обслуживания web-страниц может каждый, а при известной сноровке можно установить и использовать mod_php или mod_perl, но Apache способен на большее. Стандартная установка Ubuntu Edgy (с подключенными репозиториями Universe и Multiverse) дает перечень свыше сорока модулей Apache 2, с которыми ваш сервер Apache натворит такого, чего вы и представить не могли.

На этих четырех страницах мы расскажем вам про три наших любимых модуля *Apache* и научим работать с каждым из них. О некоторых из их функций вы, может, и слышали, но никогда не пробовали – а может, и вообще о них не знали – но запустите их в работу, и закачки пойдут быстро как никогда!

› **«Моя маман говаривала, что *Synaptic* похож на коробку шоколада» – модули с какой начинкой вы предпочтете?**

mod_userdir: Домашние страницы — народу!

Дистрибутивы, основанные на Debian, стандартно используют виртуальный хостинг, он превосходен для настройки большого числа сайтов на коммерческой основе. А если вы просто хотите выделить своим друзьям местечко на вашем сервере? Лучший способ сделать это – модуль **mod_userdir**, дающий каждому пользователю с учетной записью на вашем сервере возможность размещать свой собственный контент.

Откройте файл конфигурации **/etc/Apache2/sites-available/default** в текстовом редакторе (как суперпользователь) и добавьте такую строку:

```
UserDir public_html
```

Пользователи теперь смогут создавать подкаталог **public_html** в своем домашнем каталоге и получать к нему доступ по адресу **www.yoursite.com/~username**. У кого не будет каталога **public_html**, тот не будет представлен в Интернете – при попытке зайти на его страничку возникнет ошибка 404.

Для определения полного пути, например, **/var/www**, можно использовать директиву **UserDir**: в этом случае **www.yoursite.com/~hudzilla** будет искать страницы в **/var/www/hudzilla**. Но в принципе, лучше оставить ее в домашнем каталоге пользователя, так как это упрощает настройку прав доступа – пользователи смогут создавать и удалять

свой собственный контент в **public_html** без вашего содействия. Вы можете также использовать директиву **UserDir**, чтобы разрешать или запрещать работу модуля **mod_userdir** пользователям индивидуально. Например, сейчас каждый может публиковать пользовательские данные в личном web-пространстве, даже root – зайдите на www.yousite.com/~root, и поймете, что мы имеем в виду.

Как правило, разумнее пресечь подобный доступ, так что добавьте в конфигурацию такую строку:

```
UserDir disabled root
```

Здесь можно указать столько пользователей, сколько хотите. Вариант – настроить «белый список» **mod_userdir**, запретив доступ всем и разрешив только тем, кому нужно:

```
UserDir disabled
UserDir enabled bill ted
```

mod_cband: квоты для виртуальных серверов

Некоторые модули *Apache* могут ограничивать использование канала во времени; некоторые – ограничивать число запросов, обслуживаемое доменом в секунду; а некоторые накладывают постоянное ограничение на исходящий трафик вашего сервера. Второй из нашего трио модулей, **mod_cband**, делает все это и даже больше – что сокращает время обучения, упрощает долгосрочное администрирование и снижает требования к ресурсам вашего сервера.

Для тестирования модуля создайте текстовый файл побольше, этак на 16 МБ, набитый абракадаброй, в каталоге **/var/www**. Простейший способ это сделать – ввести команду

```
sudo dd if=/dev/urandom of=/var/www/myfile bs=16M count=1
```

Прежде чем применять ограничения, скоренько проверим, читается ли этот файл. Запустите **wget http://localhost/myfile**, и он должен загрузиться в текущий каталог. Обратите внимание на скорость, которой достигнет *wget* – наша составила 44,15 Мбит/сек, для локального файла нормально.

Откройте **/etc/Apache2/sites-available/default** как root в любимом текстовом редакторе и взгляните на директиву **DocumentRoot**. Она будет установлена в **/var/www** – нам того и надо, только добавим еще две строки прямо под ней:

```
ServerName localhost
CBandSpeed 1024 5 30
```

Именно в строке **CBandSpeed** и происходит самое интересное: она ограничивает наш сервер до 1024 Кбит/сек, при максимуме в 5 запросов в секунду и 30 одновременно открытых соединений. Если вы теперь запустите *wget* (файл конфигурации не закрывайте – он скоро вам снова понадобится), то увидите изрядное снижение скорости, поскольку **mod_cband** автоматически «душит» канал.

Ограничение использования за период

Модуль **mod_cband** вполне способен распределять ограничение ширины канала сайта во времени. Находясь в **/var/www**, запустите для начала следующие две команды:

```
sudo mkdir scoreboard
sudo chown www-data scoreboard
```

mod_cband будет хранить информацию о суммарном трафике, использованном каждым сайтом, в каталоге **scoreboard**, так что он должен быть доступен на запись пользователю, с правами которого работает *Apache* (**www-data**). Вернувшись в файл конфигурации, добавьте следующие четыре строки:

```
CBandLimit 40M
CBandScoreboard /var/www/scoreboard
CBandExceededURL http://www.linuxformat.co.uk
CBandPeriod 1M
```

Если вы точно следовали нашим инструкциям, то обнаружите, что при запуске команды *wget* никакой разницы не наблюдается. Фактически, вы можете запустить ее трижды и ничего не увидеть. Что же изменилось? А магия-то начинается с четвертой попытки: вы обнаружите, что *wget* не скачает файл **myfile** по новой, а перейдет на www.linuxformat.co.uk. Наш файл имеет размер 16 МБ, а директива **CBandLimit** настроена на запрет доступа после достижения предела в 40 МБ. Так что первое скачивание работает, поскольку уже скачано 0 МБ, второе работает, поскольку скачано 16 МБ (размер первой закачки), третье скачивание тоже работает, потому что скачано 32 МБ (сумма двух первых закачек), а вот четвертая и последующая закачки завершатся неудачей, поскольку к тому времени будет скачано уже 48 МБ, что на 8 МБ превышает лимит для вашего сайта.

Тут в дело вступает директива **CBandExceededURL**: как только будет достигнут предел трафика, любые запросы, поступающие в наш домен, автоматически перенаправятся на www.linuxformat.co.uk. Примечание: если вы попробуете это дома, то, мы уверены, осчастливите Майка, администратора сайта Команды LXF, увеличив посещаемость его детища! Хотя вы, видимо, захотите сослаться на какую-нибудь другую страницу, разъясняющую, что был превышен лимит трафика, и как можно докупить еще.

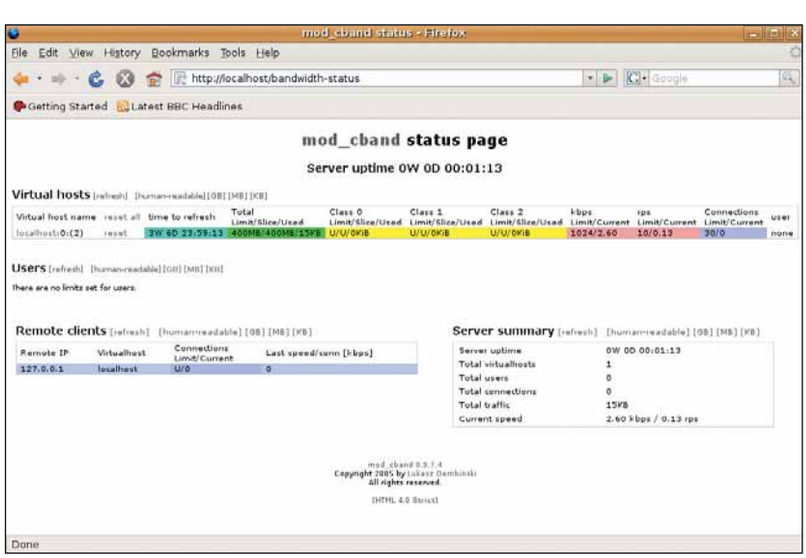
Наконец, директива **CBandPeriod** – это суммарное время между сбросами счетчиков трафика. Мы указали **1M**, то есть каждую минуту счетчик использования трафика нашего локального сайта будет сбрасываться на ноль, и пользователи смогут скачать следующие 40 МБ контента. Да, **mod_cband** использует M и для мегабайтов, и для минут, но это единственная реальная накладка. Для скоростей закачки можно »

Как установить модуль

Каждый из рассматриваемых здесь модулей доступен через *Synaptic*, если подключены репозитории Universe и Multiverse. Процесс активации установленных модулей полностью выполняется из командной строки: получите права root, перейдите в **/etc/apache2** и выполните эту команду:

```
ln -s mods-available/somemod.so mods-enabled/
```

Создастся символьная ссылка на активируемый модуль, так что для его запрета просто удалите файл-ссылку из **mods-enabled**. Активировав все модули, какие вы хотели, выполните команду **apache2ctl restart**, чтобы Apache перезапустился и изменения вступили в силу. Перезапускать Apache нужно при каждом изменении конфигурации согласно этому описанию.



› На странице статуса mod_cband дана детальная информация по всем квотам и ограничениям, но уж больно крошечным шрифтом!

» также использовать K и G (если у вас действительно быстрый интернет-канал!), а для периодов времени можно использовать S (секунды), H (часы), D (дни), и W (недели).

Есть две альтернативы директиве **CBandExceededURL**. Одна – полностью удалить эту строку, тогда **mod_cband** будет выдавать сообщение об ошибке HTTP 503 «Сервис временно недоступен». А можно применить директиву **CBandExceededSpeed**, она переопределяет **CBandSpeed** в случае превышения лимита трафика. Например, вы хотите, чтобы сайт поддерживал 1024 Кбит/сек, пока не будет достигнут порог в 5 ГБ, а затем – лишь 128 Кбит/сек до конца месяца. Вот как это можно сделать:

```
CBandSpeed 1024 5 30
CBandLimit 5G
CBandScoreboard /var/www/scoreboard
CBandExceededSpeed 128 5 30
CBandPeriod 4W
```

Одновременно использовать **CBandExceededSpeed** и **CBandExceededURL** нет смысла, поскольку они взаимоисключающие: вы либо обслуживаете страницу с ограниченной скоростью, либо перенаправляете запрос.

Отчеты о состоянии

Можете проверять состояние **mod_cband** в любое время, используя особое место на вашем сайте. Допустим, вы хотите разрешить пользователям просматривать свои лимиты и узнавать, сколько уже выбрано. Отредактируйте файл конфигурации таким образом:

```
<Location /bandwidth-status>
  SetHandler cband-status-me
</Location>
```

Вы можете теперь зайти на **http://localhost/bandwidth-status**, да и любой тоже может – советуем установить пароль на каталог, чтобы сюда попадали только свои люди.

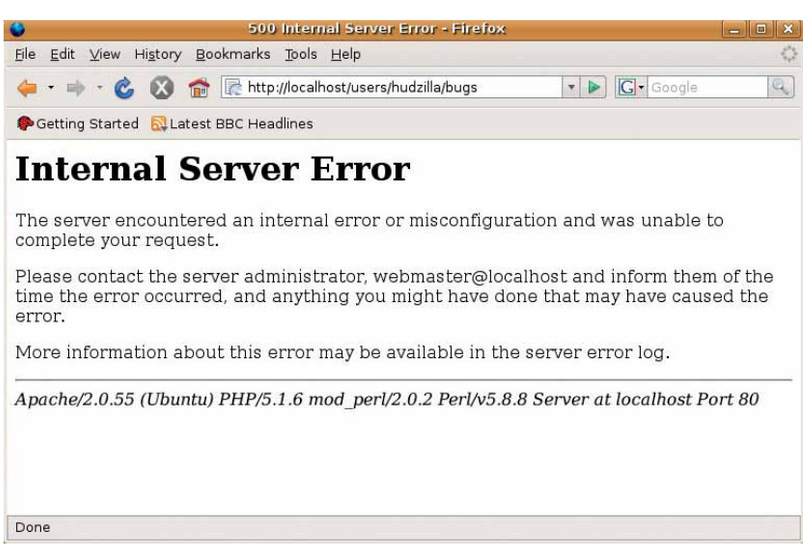
Администраторам в масштабе сервера (то есть вам), наверно, захочется видеть глобальное использование трафика, а также информацию о состоянии всех размещенных доменов. Для этого добавьте в свой файл конфигурации следующие строки для своего домена (но не для ваших клиентов):

```
<Location /bandwidth-status>
  SetHandler cband-status
</Location>
```

mod_rewrite: Перезапись URL динамически

Всякий мастер-класс *Linux Format* был бы неполон без разборки регулярных выражений, и данный – не исключение. Модуль *Apache* **mod_rewrite** придуман для проверки на соответствие строк запросов и их изменения на лету. Очень популярно использовать **mod_rewrite** для сокращения длинных URL, чтобы их легче было запоминать и набирать. Например, если вы хотите, чтобы **www.example.com/shortcut** молча перенаправлялся на **www.example.com/foo/bar/baz/wombat**, воспользуйтесь **mod_rewrite**.

Тут вы, верно, подивитесь: а чем отличается **mod_rewrite** от отсылки HTTP-заголовка Location для перенаправления браузера? Действительно, разница невелика: оба переключают один URL на другой. Однако **mod_rewrite** в общем удобнее для пользователя, поскольку заголовок Location говорит браузеру прекратить загрузку текущей страницы и вместо этого загрузить другой URL, а **mod_rewrite** ничего клиенту не сообщает, просто молча переписывает URL, чтобы тот указывал на другое место – пользователь и не заметит, что произошло.



› Работая с mod_rewrite, знайте, что «Internal Server Error» на языке Apache значит «напортачил ты в регулярном выражении, балда!»

Иначе говоря, **mod_rewrite** делает то же, что и Location, только прозрачно для конечного пользователя. Единственная издержка использования **mod_rewrite** – процессорное время, потому что *Apache* проверяет каждый входящий запрос на соответствие имеющемуся списку правил перезаписи, но это влияние довольно незначительно.

Итак, возвращаясь к примеру, где **www.example.com/shortcut** должен молча превратиться в **www.example.com/foo/bar/baz/wombat** – вот как это можно сделать с помощью **mod_rewrite**:

```
RewriteEngine On
RewiteRule shortcut/([A-Za-z0-9]*) /foo/bar/baz/wobmat/$1
```

Правило **mod_rewrite** имеет пять важных моментов:

1 Запрос на **www.yoursite.com/shortcut** покажет содержимое каталога **www.yoursite.com/foo/bar/baz/wombat**.

Почему mod_access есть зло?

Временно разрешать **mod_access**, работая с **mod_rewrite**, вполне нормально, поскольку это значительно снижает суммарное время отладки ваших регулярных выражений. Но закончив, сразу же запретите переопределение настроек, чтобы **mod_access** был капитально пресечен. Дело это хорошее, потому что **mod_access** спроектирован для поиска опций, которые следует установить в ваших каталогах, в файлах **.htaccess**. На вид безвредно, но представьте: если ваш каталог **/foo** требует пароль, как *Apache* догадается, что нужно спросить пароль, когда пользователь полезет в **/foo/bar/baz.php**?

Ответ заключается в том, что *Apache* просматривает .htaccess-файлы в запрошенном каталоге, а также во всех родительских каталогах, вплоть до корня вашей файловой системы. Так, в приведенном выше примере *Apache* ищет (и читает, если найдет) следующие файлы:

- **/var/www/foo/bar/.htaccess**
- **/var/www/foo/.htaccess**
- **/var/www/.htaccess**
- **/var/.htaccess**
- **/.htaccess**

...на что уходит уйма времени. Отключите это и используйте ваш процессор в полезных целях!

2 Запрос на www.yoursite.com/shortcut/hello отобразит файл www.yoursite.com/foo/bar/baz/wombat/hello.

3 Фрагмент **$1** ссылается на группу регулярного выражения (часть, заключенная в скобки) и использует ее в перезаписываемом URL.

4 Пользователь и не заподозрит, что просматривает **/foo/bar/baz/wombat**, а не **/shortcut** – *Apache* все делает тишком.

5 Каталог **/foo/bar/baz/wombat** по-прежнему доступен напрямую.

Теперь вы видите, чем хороша прозрачность: все, что соответствует URL www.yoursite.com/shortcut, будет автоматически перезаписано. Немного изменив правило (предусмотрев символы /, ., = и &), вы даже сможете перезаписывать целые URL, преобразуя www.yoursite.com/shortcut/gubbins/foo.php?bar=baz в www.yoursite.com/foo/bar/baz/wombat/gubbins/foo.php?bar=baz.

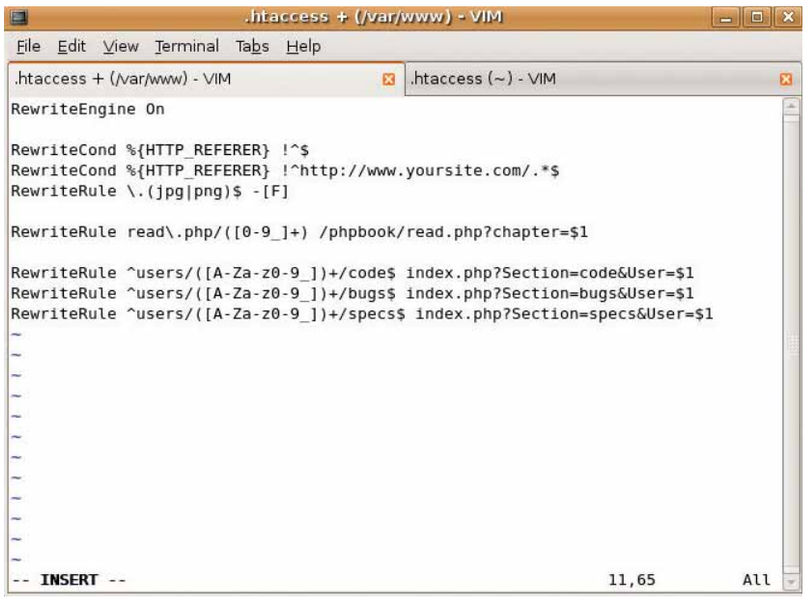
Перезапись – углубленно

Не будем заострять внимание, но ваши правила перезаписи вряд ли заработают с первого раза. Обычно требуется не одна попытка, и вы, скорее всего, добьетесь их правильности, лишь вволю наскрежетавшись зубами. Так что, добавляя новые правила, вы можете предпочесть положиться (временно!) на файлы .htaccess: они временно изменяют установки конфигурации *Apache* для конкретного каталога. Чтобы разрешить использование файлов .htaccess, загляните в конфигурацию *Apache* (ту, что мы уже используем), и для каталога **/var/www** измените 'AllowOverride None' на 'AllowOverride All'. Перезапустите *Apache*, и файлы .htaccess готовы к работе!

Пора попробовать что-нибудь поинтереснее. Да, укорочение длинного URL – типовое применение **mod_rewrite**, но еще популярнее приведение уродских URL к виду, простому для запоминания. Например, www.yoursite.com/index.php?Section=Bugs&User=Hudzilla – URL ужасный, а www.yoursite.com/users/hudzilla/bugs и запомнить легче, и для глаза приятнее! **mod_rewrite** поможет горю, например, таким образом:

```
RewriteRule ^users/([A-Za-z0-9_])+/bugs$ index.
php?Section=bugs&User=$1
```

Символы **^** и **$** означают «начало строки запроса» и «конец строки запроса» соответственно, они избавят *Apache* от обработки чего-то типа www.yoursite.com/users/hudzilla/bugs/monkeybutts.



› Использование файлов .htaccess для свежесоставленных правил mod_rewrite позволит не перезапускать *Apache* при каждом мелком изменении.

> «Опытные пользователи добавят себе мощности условиями mod_rewrite.»

Использование точных правил очень полезно, когда нужно проверять соответствие множества различных вещей. Например, вам может потребоваться загружать страницу поиска пользователя, когда на www.yoursite.com/users зайдет посетитель, а значит, строка RewriteRule должна явно игнорировать все, что добавляет имя пользователя в конец запроса. Можете написать столько строк **RewriteRule**, сколько потребуется, но помните, что *Apache* прогоняет каждый запрос через каждую из строк **RewriteRule**, и неаккуратные правила потребуют довольно интенсивной работы процессора!

Опытные пользователи могут добавить себе через условия **mod_rewrite** большей мощности. Например, если ваш сайт размещает множество изображений, и вы не хотите, чтобы другие сайты высасывали ваш трафик, устанавливая ссылки непосредственно на ваш сервер, вы можете использовать что-нибудь наподобие

```
RewriteEngine on
RewriteCond %{HTTP_REFERER} !^$
RewriteCond %{HTTP_REFERER} !^http://www.yoursite.com/.*$
RewriteRule \.(jpg|png)$ - [F]
```

Здесь три важных строки. Первая проверяет, не равен ли referrer значению **^$** – т.е. пустой строке. Учтите, что «referrer» пишется здесь с одной 'r' (**HTTP_REFERER**) из-за исторической опечатки. Если первая проверка проходит, то выполняется вторая: не совпадает ли referrer с именем нашего сайта? Это вторая строка кода. Символы **.*** в конце означают «соответствует всему», так что будет совпадение с любым URL, размещенным на нашем сайте. Наконец, если обе строки **RewriteCond** выполняются, срабатывает строка **RewriteRule**. Она отбирает JPEG- и PNG-файлы и перезаписывает их в – **[F]**, так в **mod_rewrite** сокращенно обозначено нечто невежливое вроде «запрещено – вали отсюда!».

Мы могли бы усовершенствовать второе rewrite-условие, добавив **[NC]** в конец этой строки, что велит *Apache* игнорировать регистр символов в запросе, рассматривая www.YOURSITE.com и www.yoursite.com как один и тот же путь.

Если вы озверели от безуспешной борьбы с **mod_rewrite**, можете либо расслабиться, глубоко вдохнуть и попробовать снова, либо поступить как все: взреветь от ярости и пнуть свой компьютер через всю комнату. Но что бы вы ни сделали, утешайтесь фактом, что проблемы бывают даже у профи. Брайан Белендорф [Brian Behlendorf], основатель *Apache* Software Foundation, как-то сказал: «**mod_rewrite** замечателен тем, что дает вам всю настраиваемость и гибкость Sendmail. Оборотная сторона медали – он дает вам всю настраиваемость и гибкость Sendmail.» Будьте упорны, и все получится!

Учтите: как только вы закончите с **mod_rewrite**, советуем изменить 'AllowOverride All' обратно на 'AllowOverride None': **mod_access** (модуль, который работает с файлами .htaccess) – это гарантированное узкое место производительности. LXF

Идем дальше

Если вы проработали каждый из этих модулей, вот еще три рекомендации на пробу:

» **mod_dnssd** Публикует ваш сайт, используя протокол обнаружения сети Apple Bonjour.

» **mod_musicindex** Позволяет всему миру листать вашу музыкальную коллекцию, а также и прослушивать ее.

» **mod_mono** *Apache* встречается с Mono... Разместите C# на своем сервере и запускайте ASP.NET через *Apache*.

Итак, какой модуль *Apache* ваш любимый? У вас есть трюк с **mod_rewrite**, который вам особо по душе? Напишите нам, а лучший мы опубликуем! letters@linuxformat.ru.

Лингва эзотерика

По-вашему, C – это сложно? А у Perl нехороший синтаксис? Хватит ныть, вот сейчас **Майк Сондерс** покажет вам, что такое действительно трудный язык…

Однажды Свами Прабхупада [Swami Prabhupada], основоположник движения Харе Кришна, прогуливаясь с послушниками около поля для гольфа, увидел игрока, замахнувшегося для очередного удара. «Поглядите на этого человека, – сказал Прабхупада, – всю свою жизнь он трудился, но это ничто по сравнению с усилиями, которые он прилагает сейчас, чтобы забить шар в лунку».

Такова уж людская природа: вечно нам подавай преодоление каких-то препятствий, надо или не надо. Есть же превосходные языки высокого уровня – C#, Ruby и Python, а разработчики все норовят исследовать что-то новое, и в результате появляются совсем уж причудливые (но любопытные!) создания.

В предыдущих статьях рассказывалось о Tcl, Ruby, хоть и не слишком часто применяемых, но достойных изучения хотя бы ради расширения профессионального кругозора. А сегодня мы займемся по-настоящему экзотическими языками, настолько забавными и нетривиальными, что вы увидите ремесло программирования в совершенно новом свете. Необходимые программы находятся в разделе **Magazine/Esoteric** нашего DVD.

Мы предполагаем, что вы владеете базовыми понятиями в этой области; но даже те, кто сроду кода не писал, сочтут некоторые концепции неплохой гимнастикой для ума. Вы вряд ли выберете эти языки для создания крутого пакета офисных программ, зато, изучая их, получаешь глубокое представление о строении языка программирования. Да и C после них кажется малиной…

Вьем петли из Spaghetti

Начнем с несложного языка со вкусным именем Spaghetti. В нем каждая строка кода заканчивается оператором перехода *GOTO*, причем переход на следующую строку запрещен. Иными словами, две соседние строки кода никогда не выполняются последовательно. Рациональный код и дурак напишет, а мы попрыгаем!

Почти все книги по программированию твердят, что оператор *GOTO* лучше не использовать, обходясь циклами и вызовами функций. Код Spaghetti, наоборот, петляет как заяц, на радость тем, кому циклы, функции и оператор **return** кажутся простецкими. Spaghetti – интерпретируемый язык. После запуска первые 256 байт выделенной программе оперативной памяти содержат числа от 1 до 255; есть еще

www.ellywalton-illustrations.com

26 переменных от **a** до **z**, которые находятся в ячейках с 256 по 281. Двойное слово в паре регистров **a** и **b** может использоваться как указатель на некоторую ячейку памяти. Например, установив содержимое обоих регистров равным 255 (FF в шестнадцатеричном виде), мы укажем на ячейку памяти 65535. Сокращенно этот составной адрес ячейки памяти обозначается *.

Займемся командами. Команда **a**? вводит символ и записывает его в переменную **a**, а **?a** выводит на экран значение переменной **a**. Вместо **a**, естественно, можно использовать любую переменную из диапазона **a–z**. Каждая строка кода начинается с ее номера, а заканчивается номером строки, куда нужно перейти (ни в коем случае не к следующей!). Выполнение программы начинается с первой строки и заканчивается при переходе на нулевую строку.

Оператор условия выглядит следующим образом: **a~b:5**. Если **a** равно **b**, переходим на пятую строку программы, в противном случае – на строку, номер которой указан в конце данной строки. Номера строк отделяются от собственно операторов квадратными скобками. Так, **1[a~b:3]4** означает «если **a** равно **b**, перейти к строке 3, а если нет, то к строке 4».

Пример простенькой программы:

```
2[x~13:0]3
1[x?]2
3[?x]1
```

Кристально неясно, да? Повесим на уши Spaghetti: программа просто выводит на экран символы, вводимые пользователем, и завершается при нажатии Enter. Выполнение начинается со строки 1, где с клавиатуры вводится символ, который сохраняется в переменной **x**. Затем велено перейти к строке 2 (ну нельзя же написать ее следом – она находится над строкой 1).

Тут переменная **x**, содержащая введенный символ, сравнивается с кодом клавиши Enter (13). Если **x** равно 13, переходим к нулевой строке – это, как вы помните, означает выход из программы. Если нет, переходим к строке 3 и выводим символ на экран. После этого снова возвращаемся к строке 1.

Посмотрите на текст программы еще раз, и вы получите мало с чем сравнимое удовольствие своих серых клеточек. Парочка программ на языке Spaghetti, и вы с легкостью читаете программы на Basic, которые прежде напоминали средневековые манускрипты. Ну, это… если предварительно не рехнетесь.

«Ты туда не ходи, ты сюда ходи»

Spaghetti требовал использования оператора *GOTO*, а Come Here, наш следующий язык программирования, применяет оператор COME FROM. Пусть в программе встретилось **COME FROM 30**; тогда интерпретатор после строки 30 выполнит код, следующий за **COME FROM**. Это означает, что вам нипочем не понять логику программы, пока вы не дочитаете ее до самого конца.

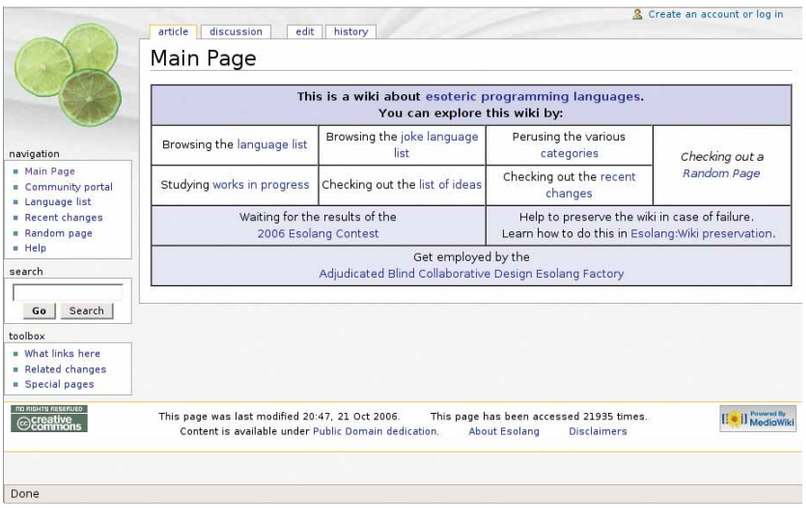
Вот программа, которая в бесконечном цикле выводит на экран слово ‘Developers’ – прямо виртуальный Стив Балмер (кстати, язык Come Here чувствителен к регистру символов):

```
COME FROM 10
10 TELL “ Разработчики” NEXT
```

Сначала выполняется строка 10, где командой *TELL* на экран выводится ‘Разработчики’. **NEXT** отмечает конец команды. Затем происходит прыжок к оператору **COME FROM 10** (раз только что была выполнена строка с номером 10), после которого выполняется опять строка 10, и так далее. Номер строки не обязателен: он играет роль метки.

Следующая программа десять раз подряд выводит на экран строку ‘Wowzers’.

```
10 CALL 10 counter
COME FROM 10 + SGN counter
TELL “Wowzers” NEXT
11 CALL counter - 1 counter
```



› Wiki про экзотические языки программирования доступна на сайте Esolangs.org.

> «Вам нипочем не понять логику программы, пока вы не дочитаете ее до самого конца.»

В строке **10** оператором **CALL** переменной **counter** присваивается значение **10**. Оператор **COME FROM** в следующей строке получает управление из строки с номером **10 + SGN counter**, где ‘SGN counter’ – это знак переменной **counter** (счетчик), равный единице, если переменная положительна. То есть пока счетчик больше нуля, из строки 11 будет делаться переход к оператору **COME FROM**. В этой строке значение счетчика уменьшается на единицу; когда оно станет равным нулю, переходов к оператору **COME FROM** больше не будет. Итого: в строке 10 устанавливается значение счетчика, затем выводится на экран строка ‘Wowzers’, в строке 11 значение счетчика уменьшается на единицу, после чего переходим к оператору **COME FROM**, строка вновь выводится на экран и т.д., до тех пор, пока значение переменной **counter** больше нуля. Просто, не правда ли?

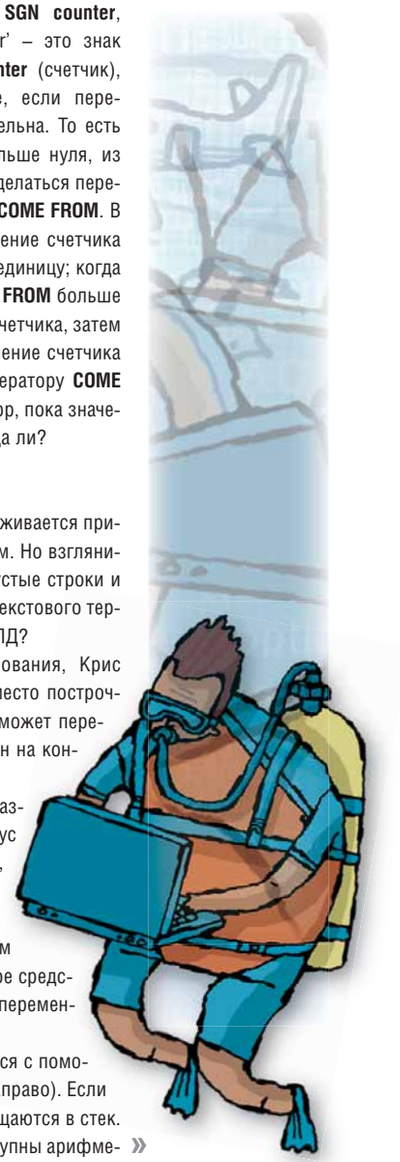
Неформальный Befunge

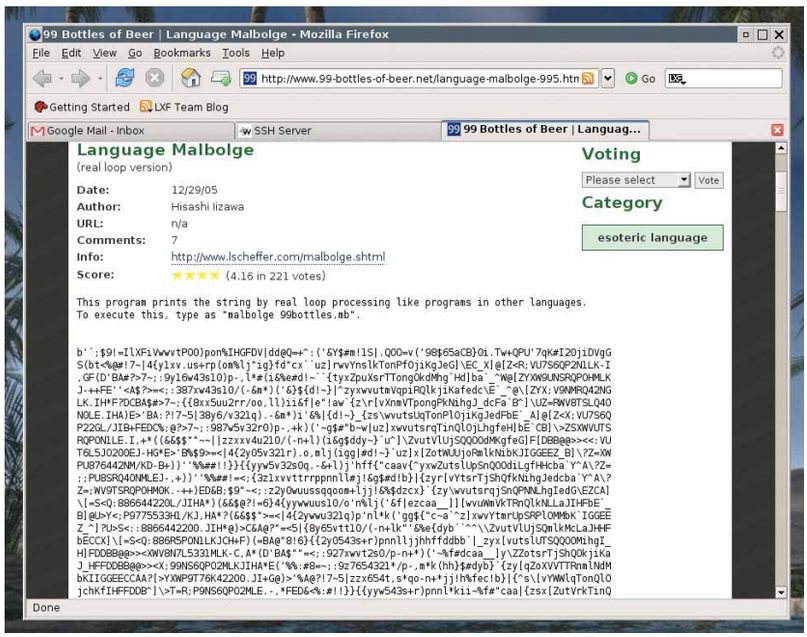
При всей своей странности, Come Here все-таки придерживается привычного размещения операторов по отдельным строкам. Но взгляните на исходный код любой программы – короткие и пустые строки и пробелы оставляют очень много места. А ведь размер текстового терминала 80x25 символов, почему бы не повысить его КПД?

Пытаясь создать язык, сложный для компилирования, Крис Пресси [Chris Pressey] в 1993 г. придумал Befunge. Вместо построчного кода, программа работает с двумерной сеткой и может переходить в любом из четырех направлений. Язык основан на концепции стека.

Напомним: стек – это блок памяти произвольного размера, учитывающий порядок записи в него. Как тубус для теннисных мячей: доступен только верхний мячик, и вложить мячи можно тоже только сверху. Например, если поместить в стек числа 3, 7 и 15, прочитаем их мы в обратном порядке: 15, 7 и 3 (последним вошел, первым вышел – по-английски сокращенно LIFO). Стек – удобное средство временного хранения данных, тем более в Befunge: переменных в этом языке нет.

Управление выполнением программы осуществляется с помощью операторов **^** (вверх), **v** (вниз), **<** (налево) and **>** (направо). Если в строке кода встречается число или символ, они помещаются в стек. Команда «запятая» (**,**) производит чтение из стека. Доступны арифме- »





› Код программы на Malbolge для песни *99 Банок Пива*, созданный компьютером. Трепещите!

тические операции **+**, **-**, **/** и ***** (сложение, вычитание, деление и умножение). Вот как выглядит программа Hello World на Befunge:

```
>                              v
v     , , , , , "Hello" <
>48*,                          v
v, , , , , , "World!" <
>25*,@
```

Попробуем разобраться. Выполнение программы всегда начинается с левого верхнего угла, там командой **>** задан переход вправо. Интерпретатор перемещается до появления следующей команды **v** и начинает движение вниз. Следующий оператор **<** вызывает движение влево, и начинается работа.

> «Malbolge — самый сложный язык программирования из существующих.»

Мы натыкаемся на строку **Hello**, и ее символы помещаются в стек в обратном порядке (olleH). Серия команд “,” по очереди считает их из стека и выведет на экран.

Затем мы движемся вниз (**v**) и направо (**>**), где помещаем в стек числа 4 и 8, вычисляем их произведение с помощью оператора * и выводим результат на экран. Результат, 32, это код символа пробела, он-то и выведется. Затем мы движемся вниз и налево. На экран выводится вторая строка **World**!.

Наконец, мы заносим в стек 2 и 5, вычисляем их произведение – 10 и выводим на экран символ с кодом 10 (переход на новую строку). Тут интерпретатор натыкается на команду завершения работы – **@**.

В Befunge еще много интересного. Ввод значений с клавиатуры и запись их в стек осуществляется с помощью операторов **&** и **~** соответственно для числовых и символьных величин. Можно поменять местами два элемента на вершине стека оператором **** или пропустить очередную команду оператором **#**. А самый захватывающий оператор – **?**, он осуществляет случайный выбор направления перехода (на все четыре стороны), в котором продолжится выполнение программы. Это может пригодиться для рандомизации программы, зато уж понять ее будет практически невозможно.

Malbolge: PHP от дьявола?

Дальше – больше. Язык Malbolge, получивший свое название от восьмого круга ада [Malebolge] из «*Божественной комедии*» Данте, считается самым сложным языком программирования за всю историю. Уж если вы его освоите, значит, вам по силам левитировать на три фута над землей или манием руки вызывать грозу. Кстати, автор выдаст 100 фунтов тому, кто напишет на Malbolge клона доктора Элизы [*Eliza*] из *Emacs* (**m-x-doctor**).

С чего это мы расщедрились? Судите сами. Первая программа на языке Malbolge появилась только через два года после его создания – да и написал-то ее не человек, а компьютер. Malbolge настолько сложен, что простые смертные справляются в лучшем случае с выводом строки текста. Даже программы, созданных компьютером, на большее особо не замахиваются. И это плохо, скверно, не здорово.

Интерпретатор языка Malbolge представляет собой тернарный процессор. Если вы программировали на языке Assembler, то знакомы с двоичной (основание 2) и шестнадцатеричной (основание 16) системами счисления. Троичная система счисления исключительно неуклюжа, а Malbolge того и надо. Посчитайте: 0, 1, 2, 10, 11, 12, 20, 21, 22, 100… Ужас.

Но это еще цветочки. Пользователю доступны три регистра (аналог переменных) **A**, **C** и **D**. Регистр A – аккумулятор, он хранит результаты выполнения операций; регистры **C** и **D** содержат адреса ячеек памяти для кода и данных соответственно, причем после выполнения очередной команды их содержимое увеличивается на единицу. Доступны 59049 ячеек памяти, каждая из которых может содержать десятиразрядное троичное число.

Восемь основных команд языка включают команды перехода, троичного сдвига, ввода/вывода и завершения программы. Есть еще «шальная» команда, по двум операндам выбирающая значение из таблицы – так Malbolge заполняет неиспользованную память по данным использованной.

Когда интерпретатор начинает выполнение очередной команды, к содержимому регистра **C** (скажем, 4) прибавляется смещение этой команды в коде программы (например, 390). Затем из полученного числа вычитается 94 до тех пор, пока результат больше 94. Результат этой операции определяет код выполняемой команды. После выполнения команды содержимое соответствующей ячейки памяти шиф-

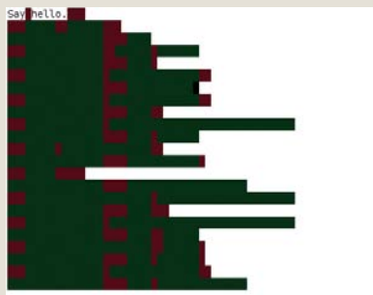
Здесь смотреть нечего

В статье про экзотические языки нельзя не упомянуть о Whitespace. Этот язык программирования, команды которого состоят из пробелов, символов табуляции и перевода строки, не видимых на экране, по праву можно считать самым чудаковатым. Открыв его программу в текстовом редакторе с подсветкой синтаксиса, вы увидите одни разноцветные пятна.

Тем не менее это полноценный язык. Например, команде вычитания соответствует последовательность из символа табуляции, двух пробелов и еще одного символа табуляции. Как и в языке Befunge, широко используется стек. Вы можете создавать процедуры и пользоваться метками. Учтена даже модульность, столь важная для крупных проектов!

Рассмотрим пример программы:

Здесь мы складываем два числа и записываем результат в память. Разработчики уверяют, что язык очень популярен для приложений с высокой защищенностью: даже распечатав текст программы, злоумышленник не увидит в ней уязвимостей. Кстати, распечатка сберегает окружающую среду, экономя на чернилах. Чем не язык для пост-структуралистов?



› Программы на Whitespace видимы только в редакторах с подсветкой синтаксиса.

руется при помощи таблицы подстановки, и переход на эту ячейку станет невозможен.

Вот и все, что мы можем открыть, а не то журнал вспыхнет у вас в руках – о прочей черной магии узнаете через ссылки в нашей врезке Ресурсы, ниже на этой странице. В заключение – программа, выводящая на экран строку “Hello, world” на языке Malbolge:

```
(=<`:9876Z4321UT.-Q+*)M'&%$H"!~}|Bzy?=|{z]KwZY44Eq0/{mlk**
hKs_dG5[m_BA{?-Y;;Vb'rR5431M}/.zHGwEDCBA@98\6543W10/.
R,+O<
```

Нигилизмы

Радуйтесь: даже в мире экзотических языков программирования бывают передышки на что-то менее безумное. Разработчики Nil применили свежий подход к программированию, поставив дизайн впереди программных ухищрений. Много говорено, что языки высокого уровня типа C# избавляют от рутинных операций, позволяя сосредоточиться на алгоритме, а Nil превращает этот процесс в настоящее искусство.

Nil дает существенное уменьшение размера получаемого объектного кода, упрощение отладки программы и полную межплатформную переносимость. Это один из немногих языков, которые, пообещав эти преимущества, в самом деле обеспечивают и их, и многое другое!

Формат исходного кода языка может варьироваться. Рассмотрим фрагмент кода:

```
for(b = 1; bytes < c_byte; b++) {
    if(!(bh[b] = sb_getblk(s, ++cur_index)))
        GOTO block_release;
    bytes += msBlk->devblksize;
}
```

Здесь содержатся операторы цикла и сравнения. Как этот код оптимизировать? Компилятор языка Nil расщепляет программу на «атомы», удобные для синтаксического анализа. Каждый «атом» транслируется отдельно; Nil практикует нигилистический подход, согласно которому «атомарная» операция есть ни что иное как ничто, т.е. **nop**, и выполняет соответствующие подстановки.

С позиций данной оптимизации, Nil сворачивает последовательность команд **nop** в единственную команду **nop**, существенно уменьшая время выполнения программы и размер используемой памяти. Команда **nop** поддерживается подавляющим большинством процессоров, поскольку рекомендует им не делать ничего. Итак, код приведенного фрагмента кода на языке Nil будет выглядеть следующим образом:

```
nop
```

нигилисты ведь утверждают, что все на свете бессмысленно! Зачем процессору париться с каким-нибудь преобразованием Фурье? Пусть постынет, пока вы пьете чаек!

В заключение

Таковы подлинно экзотические языки программирования. Может, они и не созданы для величия, но все равно волнуют умы. Забавно попробовать Spaghetti, хотя бы для того, чтобы взять да и усложнить простые вещи – это тренирует мозги; а после головоломного Malbolge вы вернетесь к программированию на добром старом C с новым энтузиазмом, как скороход, сбросивший с ноги гирю.

Во врезке Ресурсы вы найдете ссылки на сайты с дополнительной информацией. Испробуйте эти языки сами и обязательно расскажите нам об этом! LXF

C: чего только не придумают

Одна из бед языка C заключается в том, что здесь не выработан стандартный стиль программирования (в отличие, например, от Python). На этом языке вполне можно писать читаемые программы, но уж слишком легко все запутать отступами, пробелами, именами переменных, макросами и прочим. Если вам приходилось разбираться в чужом коде, вы наверняка не раз проклинали все на свете, недоумевая: «Ну что он хотел этим сказать?».

К счастью, теперь вам есть куда послать нечитаемый код. Каждый год участники конкурса International Obfuscated C Code Contest (IOCCC, Международный конкурс на самый запутанный код) выбирают лучшие образцы кода из тех, которые работают совсем не так, как должны на первый взгляд. Некоторые программисты создают настоящие произведения ASCII-искусства – даже в голову не придет, что это программа, которая корректно компилируется.

Взгляните на рисунок. На вид это просто портрет, а на самом деле – исходный код; запустите его, и он выведет на экран другую картинку. На сайте конкурса www.ioccc.org таких программ много. Мы уверены, что даже настоящие гуру языка C не с ходу в них разберутся. Зато как весело скомпилировать такой пример и посмотреть, что будет...

› Запустите эту программу – получится… другая картинка.

Ресурсы

» http://kidsquid.com/files/spaghetti/spaghett.txt Информация по Spaghetti.

» http://en.wikipedia.org/wiki/Befunge Домашняя страница Befunge.

» http://tinyurl.com/y8ta2z Come Here.

» www.antwon.com/other/malbolge/malbolge.txt Углубляемся в Malbolge…

» http://compsoc.dur.ac.uk/whitespace … и Whitespace.

» www.ioccc.org Конкурс IOCCC.

» http://esolangs.org/wiki/Main_Page Разыщите еще что-нибудь сами.

Что за штука... OpenID?

Вы уже перестали понимать, кто вы такой? Все, что вам нужно – это URL, утверждает **Марко Фиоретти**.

Что такое OpenID?
Если верить лозунгу с домашней страницы OpenID (http://openid.net), это – лучшее изобретение человечества со времен колеса: «открытая, свободная, распределенная система поддержки цифровой личности, ориентированная на пользователей».

И как понимать слова «цифровая личность»?
Грубо говоря, это любой программный метод и набор цифровых данных, позволяющие системе выяснить (без какой-либо неопределенности и дыр в безопасности), кем кто-то или что-то является на самом деле.

Но я совершенно не хочу, чтобы кто-то выяснял, кто я такой! Интернет-свобода существует исключительно благодаря анонимности!

> “Чтобы интернет стал действительно полезным, ему необходима открытая, распределенная и вездесущая система цифровой личности..”

Ну, начнем с того, что действительная анонимность в Интернете – это всего лишь иллюзия. Для того, чтобы действительно замести все следы, нужно предпринять много действий, часть из которых иногда противоречит договору с интернет-провайдером, а то и просто незаконна. Но даже при этих условиях бывает множество ситуаций, в которых вы сами хотите или должны дать знать интернет-партнеру о том, кто вы такой.

Например?
Например – интернет-банкинг. Для вас жизненно важно, чтобы банк проверил, кто вы такой. Да и вам было бы полезно проверить, что вы в своем банке, а не в одном из широко расплодившихся мошеннических сайтов. Существует множество ситуаций, в которых возможность быстро идентифицировать всех интернет-участников может сэкономить много усилий и времени.

Некоторые люди уже используют OpenID, чтобы дать возможность друзьям читать их закрытые публикации в LiveJournal, не проходя регистрацию на этом сайте.

Гмм... Ну, пускай. Но кто контролирует OpenID? Правительство? Злобные инопланетяне? Или еще какой-нибудь Большой Брат?
А вы помните «ориентировку на пользователя», которую я упоминал в самом начале? OpenID зиждется на том, что каждый может идентифицировать себя самостоятельно, а также предоставить сервис цифровой идентификации для других. Конечно, благонадежность какого-то из самодельных провайдеров может вызывать вопросы, но зато никто не завязан на одну-единственную организацию.

И как работает OpenID?
Если вы хотите узнать все подробности, не вникая при этом в технический жаргон – прочитайте пошаговое описание (с красивыми картинками), опубликованное на www.openidenabled.com/openid/openid-protocol. Ну, а я попробую показать вам на пальцах.

Как мило. А я по-прежнему должен буду помнить все эти имена, пароли и вымышленные девичьи фамилии бабушки?
Нет. OpenID использует всегда один и тот же URL, обычный интернет-адрес, вне зависимости от того, на каком сайте вы хотите заявить о себе. Ваш пароль и любая другая идентификационная информация, которой вы согласны поделиться, известна только выбранному вами серверу OpenID.

Прекрасно, выходит, мне нужно установить и обслуживать еще одно приложение, работать с которым могут только хакеры.
Не обязательно... Вы, конечно, можете запустить свой собственный сервер OpenID у себя дома, но можете воспользоваться и одним из сторонних провайдеров. Их список есть на www.lifewiki.net/openid/OpenIDServers.

Ну что ж, до сих пор все звучит неплохо. Как мне воспользоваться OpenID?
Если вы просто хотите выступить в качестве потребителя, то есть идентифицироваться при помощи OpenID на поддерживающих его сайтах (не интегрируя при этом его поддержку в свой собственный сайт), то все, что вам нужно – это получить свой идентификатор. Их существует два типа – URL и XRI (eXtensible Resource Identifiers, расширяемый идентификатор ресурса). Самый быстрый и простой способ получить URL – это регистрация на одном из общедоступных провайдеров OpenID. При этом вы получите адрес, обычно в виде http://yourname.someOpenIdprovider.com, уже настроенный на представление OpenID.

Это значит, что я не могу использовать свое собственное доменное имя как свой OpenID URL, если я предварительно не настроил на нем OpenID-сервер?
Конечно, можете! Между прочим, сделать вашу домашнюю страницу вашим OpenID URL – это гораздо круче и профессиональнее и не требует большого труда! Для начала зарегистрируйтесь на каком-нибудь провайдере OpenID. После чего добавьте в секцию <head> вашей домашней страницы несколько строк (подробная инструкция находится по адресу www.openidenabled.com/openid/use-your-own-url-as-an-openid) – и готово! Любой, кому потребуется идентифицировать вас по OpenID, найдет на вашей домашней странице указание, с каким сервером для этого нужно связаться.

Пусть у меня уже есть нужный URL. Что мне с ним делать, когда я хочу доказать, что я – это я?
Чтобы зайти на любой сайт с поддержкой OpenID, найдите нужную форму и укажите в ней свой URL. Сайт использует его, чтобы связаться с вашим провайдером OpenID, и автоматически перенаправит вас на форму входа, расположенную на нем. После того, как вы пройдете аутентификацию на сервере OpenID, ваш провайдер сделает две вещи, причем опять же автоматически. Во-первых, он передаст подтверждение на тот сайт, с которого вы изначально делали запрос, что вы – это действительно вы, и с этого момента вас надо рассматривать как полноценного зарегистрированного пользователя системы. Во-вторых, провайдер перенаправит вас на ту самую страницу, с которой вы начинали.

Если мне по-прежнему придется вводить пароль, то в чем тогда смысл OpenID?
Смысл в том, что теперь вам надо помнить всего один URL и всего один пароль, вне зависимости от того, сколько сайтов вы посещаете. В зависимости от политики сайта, ваш провайдер может запросить у вас разрешение на передачу каких-то дополнительных персональных данных. Сейчас разрабатывается система, позволяющая делать то же самое автоматически, еще проще и быстрее. И, конечно же, вас всегда спросят, устраивает ли вас доверяемый корневой элемент (trust root) web-сайта, который пытается идентифицировать вас.

Эз... Простите, доверяемый что?
Доверяемый корневой элемент (ДКЭ) – это строка, похожая на обычный URL, которую сайты, поддерживающие OpenID, должны отправлять вместе с каждым запросом на вашу идентификацию. ДКЭ позволяет добиться того, что ваш провайдер OpenID не будет автоматически подтверждать вашу личность для

всех сайтов на свете. Сначала (когда вы будете перенаправлены на сервер OpenID) он должен спросить вас, согласны ли вы с тем, чтобы запрос на авторизацию автоматически подтверждался для всех доменов и поддоменов адреса, приведенного в качестве ДКЭ. Конечно, если вы выполняете вход на сервер www.someonlineshop.com, а провайдер OpenID запрашивает у вас согласие на вход в www.westealcreditcards.com, соглашаться – не самая лучшая идея. Никогда не подтверждайте ДКЭ, URL которых не совпадает с адресом посещаемого вами сайта!

Какова оборотная сторона медали OpenID?

Точно такая же, как и у любого доменного имени в Интернете, или, если хотите, у телефона или домашнего адреса – к сожалению, никто не гарантирует, что они останутся неизменными на протяжении вашей жизни. И, точно так же, опрометчиво надеяться заполучить адрес, состоящий из вашего имени и фамилии, особенно если вас зовут Иванов Иван Иванович.

Неужели эта проблема неразрешима?

Ну почему же... Расширенные идентификаторы ресурсов, которые я упоминал выше, решают проблемы постоянного адреса, так как они всегда состоят из двух частей. Первая часть называется i-name – это просто строка, которую несложно запомнить и легко передать кому-то другому. Но когда кто-нибудь запрашивает i-name в первый раз, он всегда выдается одновременно с еще одним идентификатором, i-number, который уже никогда не может быть выдан кому-то еще. В дальнейшей работе i-name выступает только в качестве синонима i-number. Такая схема позволяет, кроме всего прочего, запросить новый i-name для существующего i-number, когда старый становится недействительным (чаще всего такое происходит, когда меняется название бизнеса).

Где я могу найти онлайновую документацию?

На официальном сайте OpenID и в списках рассылки, расположенных по адресу http://openid.net/mailman/listinfo. В них рассматриваются вопросы безопасности, впечатления пользователей и так далее.

А что если я захочу добавить поддержку OpenID в мое web-приложение?

Замечательно! Зайдите на www.lifewiki.net/openid/OpenIDLibraries и возьмите оттуда готовый код. Уже существуют интерфейсы для использования его из PHP, Perl, Python, Java и многих других языков программирования.

Не сомневаюсь, что им требуются разработчики...

Так и есть. Если вы не против запачкать руки, заходите на www.openidenabled.com. Кроме того, вы можете побороться на http://iwantmyopenid.org/bounty за приз, предназначенный для тех, кто интегрирует OpenID в приложения FOSS. Но разработчикам нужен не только код. Если вы чувствуете, что в вас проснулся проповедник OpenID, срочно идите на http://iwantmyopenid.org!

Что будет с OpenID дальше?

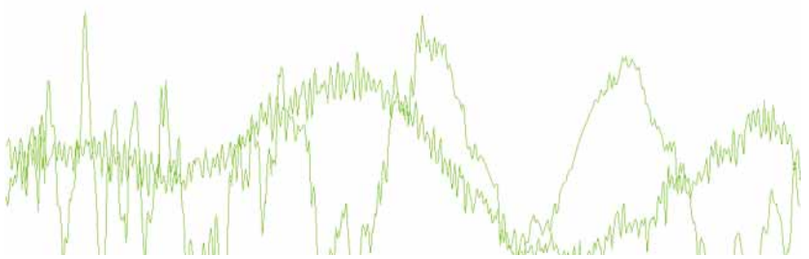
Сейчас идет работа над OpenID Authentication 2.0. Это обновление спецификации ключевой функции OpenID – аутентификации пользователей. Еще одна новая спецификация, OpenID Attribute Exchange, дает пользователям возможность указывать, каким образом провайдер должен делиться элементами личной информации (телефоном, адресом) с третьими лицами.

Как вы думаете, это будет работать, или это просто еще один мыльный пузырь?

Посмотрите на список сайтов, которые уже поддерживают OpenID, на www.myopenid.com/directory. В него входят Technorati и Movable Type! Чтобы Интернет стал действительно полезным, ему необходима открытая, распределенная и вездесущая система поддержки цифровой личности. Когда OpenID превратится в такую систему – это уже другой вопрос. Но если он будет интегрирован в такие продукты, как WordPress и Drupal, он может очень быстро стать стандартом de facto перед тем, как эту нишу займет кто-нибудь другой, чем поможет нам избежать множества проблем в будущем. LXF

Тяжелая

ЧАСТЬ 3 Петр Семилетов готовится рассказать вам о программах настолько высокого класса, что их название сложно перевести на русский язык короче, чем «рабочая станция по цифровой обработке звука».



Сегодня мы подобрались к «тяжелой артиллерии» музыкальных программ – к той их нише, которая во всем мире кратко называется DAW (Digital Audio Workstation), то есть рабочая станция по обработке цифрового звука. В русском я не знаю подходящего термина, и чтобы не писать все время «DAW», буду употреблять словосочетание «рабочая станция».

Изначально DAW-ом называлась связка из программы, позволяющей делать многоканальную запись, и дорогой звуковой карты под эту программу (или наоборот – программы под карту). Примеры – Protocols, Nuendo. Сейчас функциональность типичной рабочей станции несколько шире – это не только запись и микширование звуковых дорожек, но и встроенный MIDI-секвенсер с поддержкой виртуальных инструментов. Есть «чистые» DAW-ы, а есть смешанные – со встроенным MIDI-секвенсером.

DAW – это сердце домашней/любительской студии. Сюда поступают все партии, здесь же они и микшируются. Рабочие станции – пожалуй, самые сложные музыкальные программы. Сложные как внутренне, алгоритмически, так и для пользователя. Хотя все DAW'ы имеют ряд сходных составляющих, как то: монтажный стол, где располагаются партии/сэмплы и отдельным окном плавает виртуальный микшер. Желательны также встроенный (хотя бы простейший) волновой редактор, отображение видео (для синхронизации звука с видеорядом) и автоматизация – то есть запись и воспроизведение динамических изменений параметров эффектов. Пример – управление громкостью дорожки, чтобы она не находилась постоянно на одном уровне, а в нужных местах становилась меньше или больше.

В этой статье мы посмотрим на две Linux-программы из разряда рабочих станций – *Rosegarden* и *Muse*. Есть и другие – *Ardour* (см. **LXF89**), *Audacity*, также хороша в работе с цифровым звуком *Cinelerra* (см. **LXF85**). *Rosegarden* и *Muse* относятся к числу того универсального ПО, который сочетает в себе как функции «классических» DAW (т.е. без поддержки MIDI), так и MIDI-секвенсера. К слову сказать, в *Rosegarden* упор сделан именно на MIDI и сопутствующие технологии (виртуальные синтезаторы и т.д.).

Rosegarden и *Muse* выбраны мною потому, что это программы примерного одного класса ПО, и приблизительно одного уровня. Было бы негоже сравнивать чистый DAW вроде *Audacity* с *Muse*. Кроме того, *Rosegarden* и *Muse* похожи между собой. Обе программы слывут «нашим ответом *Cubase*» (это мы обсудим чуть позже), интерфейс обеих так или иначе основан на *Qt* (в *Rosegarden* имеется привязка к KDE). Итак, приступим.

Rosegarden

Web www.Rosegardenmusic.com

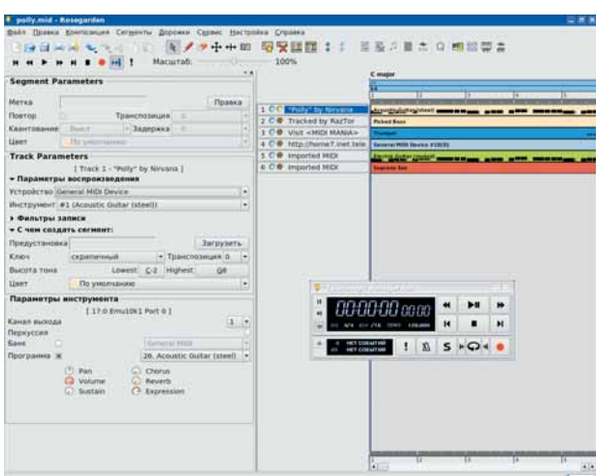


Рис. 1.

На сайте продукта можно найти сравнение с *Cubase*: ««the closest native equivalent to *Cubase*® for Linux» – Sound on Sound». Некоторое сходство и впрямь есть – монтажный стол справа, свойства дорожек слева, и плавающее окно «Транспорт». Да, вот вам и первое отличие – *Rosegarden* русифицирован, а официальных переводов *Cubase* на русский, насколько я знаю, нет. Других отличий больше, нежели сходства, поэтому сравнивать с *Cubase* я (как завзятый реалист) перестаю.

Свежий (во время написания статьи – версии 1.5.0) *Rosegarden* без труда устанавливается из исходных текстов на современный «бытовой» дистрибутив вроде моего Mandriva 2007, то есть никаких запредельных зависимостей от библиотек не предполагается.

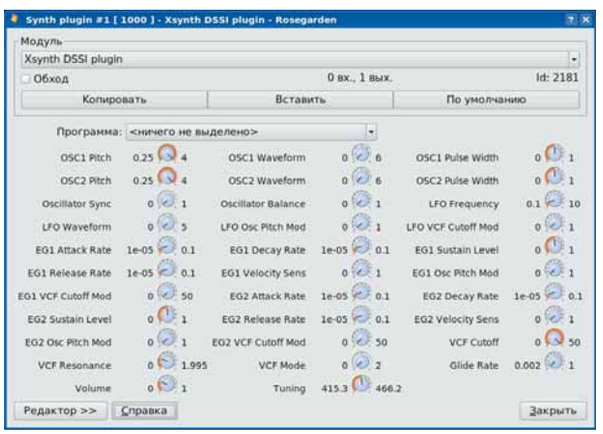
Для сборки используется модная нынче утилита *Cmake*. Однако, я советую установить более раннюю версию *Rosegarden* – например, 1.4.0 или любую после 1.5.0 (если в ней исправят импорт WAV/MP3/OGG-файлов). В 1.5.0 такой импорт попросту сломан. Во всяком случае, мне так и не удалось заставить его работать.

Для сборки *Rosegarden*, кроме заголовочных файлов KDE, требуется много чего еще, в том числе – библиотека DSSI (http://dssi.sourceforge.net). DSSI (произносится как «диззи») – это такое API программных синтезаторов, аналог VSTi из мира Windows. Если не ошибаюсь, DSSI придуман тем же разработчиками, которые создают *Rosegarden*.

Месяц назад Мы повстречались с сервером Jack и познакомились с барабанными машинами.

артиллерия

Что дают DSSI-инструменты нам, пользователям? Возможность расширить палитру своих инструментов. DSSI-инструмент – это модуль, плагин. Вы загружаете его в *Rosegarden* (или другую программу) и используете его как инструмент, на который можно, грубо говоря, направлять MIDI-дорожку, т.е. посылать на него MIDI-команды и данные.



› Рис. 2. DSSI-синтезатор *Xsynth*.

На сайте DSSI, помимо общих сведений и самой API-библиотеки, есть список DSSI-плагинов, покамест не очень большой. Но и там есть из чего выбрать. Я рекомендую следующие:

›› Nekobee – очень приблизительное виртуальное воплощение знаменитого синтезатора TB-303.

›› Hexter – моделирует синтезатор Yamaha DX7. Есть 70 предустановленных патчей.

›› XSynth – просто довольно навороченный синтезатор.

›› Fluid Synth – сэмплер, который умеет загружать звуковые банки формата Sound Fonts. Скачать такие банки бесплатно можно с http://www.hitsquad.com/smm/win95/SOUND_FONTS/ и http://www.personalcopy.com/sfonts.htm.

Несколько замечаний о запуске *Rosegarden*. Перед работой с ним надо запустить звуковой сервер JACK. Делается это примерно такой командой:

```
jackd -d alsa
```

Конечно, вы можете добавить опции на свой вкус, либо, если ваша звуковая карта работает через OSS, заменить «alsa» на «oss». Команду запуска JACK можно также прописать в Настройках программы, на странице «Секвенсер – Запуск», в поле «Команда для запуска JACK». И поставьте галочку на «Запускать JACK при загрузке *Rosegarden*».

Кроме того, на вкладке «Общее» можно настроить загрузку банка Sound Font при запуске программы. Для этого используется внешняя утилита *sfxload*. Сам банк можете взять либо на приведенных выше сайтах, либо у себя на Windows-разделе, если таковой имеется и вы используете звуковую карту Creative. В этом случае у вас должны быть такие банки: **8mbgmsfx.sf2**, **4gmgsmt.sf2** и **2gmgsmt.sf2**. В каждом из них – набор патчей для инструментов по таблице General MIDI, то есть все 128 инструментов. Эти банки отличаются размерами. Лучше всех звучит, конечно же, восьмимегабайтный **8mbgmsfx.sf2**.

При запуске *Rosegarden* может обнаружить, что ваш системный таймер имеет слишком низкое разрешение для хорошей работы с MIDI. Это разрешение зависит от параметров ядра (Processor Type and Frequency > System Timer, рекомендую значение 1000 Hz) – по умолчанию там может стоять слишком маленькое (для потребностей программы) значение. В ядрах эдак с версии 2.6.9 значение уже достаточно велико, чтобы удовлетворить *Rosegarden*.

Теперь посмотрим на рабочие качества *Rosegarden* по двум категориям – работа с MIDI и работа с цифровым звуком. Точно так же мы пройдемся и по *Muse*.

Работа с MIDI

Упор в *Rosegarden* сделан именно на MIDI. Обработка цифрового звука развита гораздо меньше, но об этом – чуть позже. Редактировать MIDI-дорожки можно в четырех встроенных редакторах. Это «Редактор событий» (предоставляющий самый низкоуровневый доступ к MIDI-данным), «Редактор перкуссии», «Матричный редактор» (пианоролл) и нотный редактор.

Редактор перкуссий отличается от пианоролла разве что списком с названиями ударных инструментов, что облегчает ввод нот – сразу видно, где какой инструмент. Напомню, что для ударных в MIDI отведен канал номер 10 – его и надо выставить в свойствах дорожки (здесь номера каналов доступны в списке «Инструмент») (**Рис. 3**).

Кстати, об этих свойствах. Как и в *Cubase*, свойства дорожки расположены в главном окне слева от нее. Из любопытных свойство отмечу следующие. Можно включить транспозицию, квантование и задержку – как бы эффекты реального времени. В выпадающем списке можно выбрать цвет, которым будут отображаться на дорожке данные MIDI-партии. В случае использования DSSI-инструмента, в свойствах дорожки можно вызвать редактор настроек для этого инструмента, а также «навесить» на инструмент до пяти LADSPA-эффектов.

› Рис. 3.

Пианоролл – матричный редактор. Почему-то по умолчанию выключены элементы управления MIDI-контроллерами. Согласен, используются они нечасто. Хотя в Sonar или *Cubase* они постоянно видны, и сразу становится ясно, что в наличии такая функция есть, а в *Rosegarden* надо еще пойти в меню Вид (в Матричном редакторе) и там включить нужное (**Рис. 4**).

Управление контроллерами реализовано довольно непривычно. Их нельзя рисовать карандашиком или стирать ластиком. Вместо

Технологическая справка

Аппаратной способностью воспроизводить Sound Fonts обладает только линейка звуковых карт от Creative, начиная с AWE32. В AWE32 и AWE64 звуковые банки загружались в оперативную память самой звуковой карты. В картах Live! и выше используется уже оперативная память компьютера. Для Fluid Synth не имеет значения, какая у вас звуковая карта. А звуковые банки формата Sound Fonts можно делать и самому.

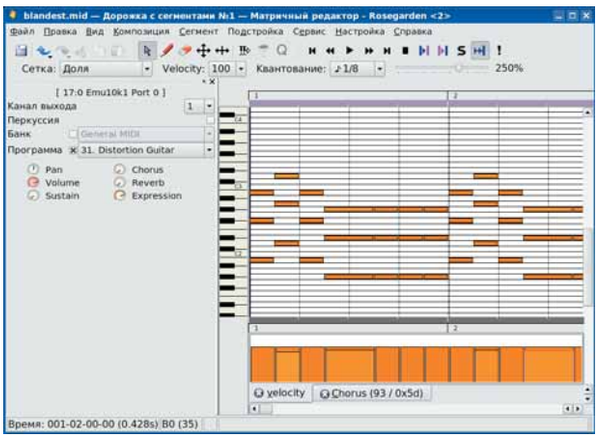


Рис. 4.

этого нам предлагается работать с эдакими столбиками, которые именуются объектами. Добавляются они через контекстное меню, а уровень высоты каждого столбика задается мышью.

Нотный редактор – именно он открывается в *Rosegarden* для редактирования MIDI по умолчанию. Понятно, почему так происходит – нотный редактор (особенно такой, как в *Rosegarden*) придает программе вес. Я не знаток нотной грамоты, но в *Rosegarden* вполне можно верстать нотные листы, причем с текстами. Вот бы еще скрестить *Rosegarden* и *KGuitar* (Linux-аналог *Guitar Pro*) и иметь возможность рисовать аппликатуры аккордов! Впрочем, у разработчиков *Rosegarden* есть в планах некие дополнительные функции по гитарной части. Кстати, из *Rosegarden* можно экспортировать партитуру в формат известной программы верстки нот – Lilypond (**Рис. 5**).

Чем еще хорош нотный редактор, так это своим видом – ноты отображаются на фоне эдакой старой дешевой бумаги, на которой, должно быть, приятно писать чернильной ручкой.

Что до «нотных» шрифтов, то три таких шрифта входят в дистрибутив *Rosegarden* и устанавливаются вместе с программой автоматически. Отдельно эти шрифты устанавливать не нужно (хотя, при желании, можно).

Работа с цифровым звуком

Подразумевается работа с, грубо говоря, wav-файлами, а вернее – с дорожками, на которых располагаются сэмплы. Реализовано это довольно оригинальным образом. Чтобы записать звук со входа звуковой карты на дорожку, надо включить на ней красную кнопочку (запись), затем пустить композицию на воспроизведение, и уже по ходу этого воспроизведения нажать на другую красную кнопку – на панели транспорта. В других программах, кнопку записи обычно надо нажимать до «воспроизведения». Это и сбивает с толку.

Записанный фрагмент появляется на дорожке, а кроме того, становится доступен из окна «Звуковые файлы проекта» (вроде инструмента Pool в *Cubase* VST). К сожалению, встроенного волнового редактора в *Rosegarden* нет, хотя можно вызывать внешний (задается в настройках).

Rosegarden 1.5.0 по какой-то причине перестал импортировать внешние WAV-файлы, впрочем, та же участь постигла MP3 и OGG. Чем это объясняется – мне неведомо, но в версии 1.4.0 (с теми же библиотеками) всё работало.

На аудио-дорожку, как и на виртуальный синтезатор, можно навесить до пяти LADSPA-плагинов. На этом, собственно, работа с аудио-дорожками исчерпывается, хотя нет – есть еще «растягивание» звука, time-stretching, которое осуществляется инструментом «Изменить размер» с зажатой клавишей Ctrl. Растягивание это опасно тем, что с каждой выполненной операцией по растягиванию создается новый файл с измененными звуковыми данными. На мой взгляд, такая реализация функции – не самая удачная.

К сожалению, нет для звуковых дорожек и таких привычных вещей, как огибающих (или кривых) громкости и панорамы. Вы можете настроить громкость и панораму лишь статично, через микшер или в свойствах дорожки. Каких-либо признаков автоматизации (динамическое изменение параметров эффектов) тоже нет. В этом плане *Rosegarden* подобен классическому «железному» микшерному пульту.

MusE

Web **www.Muse-sequencer.org**

К сожалению, установить новейшую версию (1.0 alpha release) этой программы мне не удалось – ей нужна *Qt 4.2*, а у меня была только *Qt 4.1*, поэтому рассказывать буду, опираясь на опыт общения с *Muse 0.9pre2* – она требует старый добрый *Qt 3.x*.

Что до новшеств в *Muse 1.0*, то могу их назвать – это поддержка модулей DSSI и VST, MIDI-плагины, графический редактор автоматизации, и многое другое. Автоматизация есть и в предыдущих версиях, но редактировать ее можно только передвижениями/вращением всяких элементов управления.

Для установки *Muse 1.0*, как и в *Rosegarden*, используется утилита *Cmake*, а в версии *0.9pre2* – стандартные autotools.

В дистрибутив *Muse* входит несколько виртуальных синтезаторов (не DSSI): навороченный, со множеством пресетов DeicsOnze, драммашина SimpleDrums (правда, без сэмплов – их надо загружать извне), симулятор органа Organ 1 и другие (**Рис. 6**).

Для нормальной работы *Muse* требуется запущенный JACK-сервер. Для карт Creative также можно подгрузить звуковой шрифт командой *sfxload*.

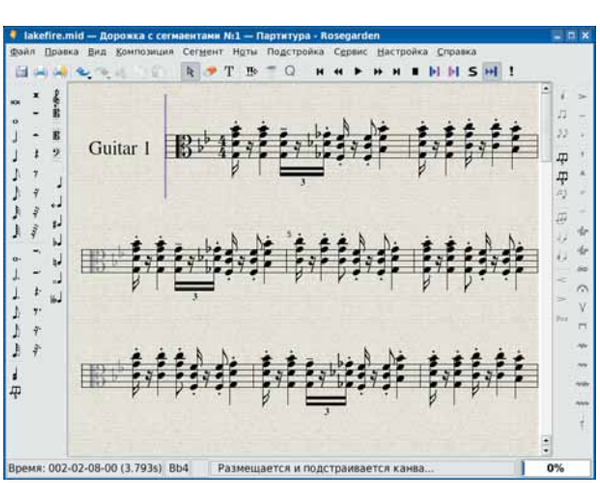


Рис. 5. Вот так выглядит хороший нотный редактор.

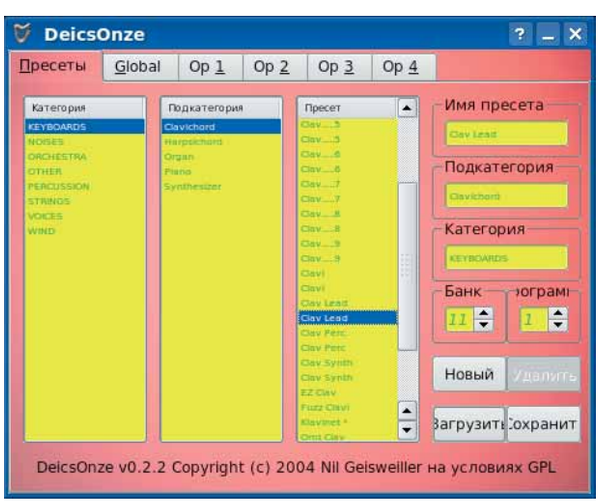
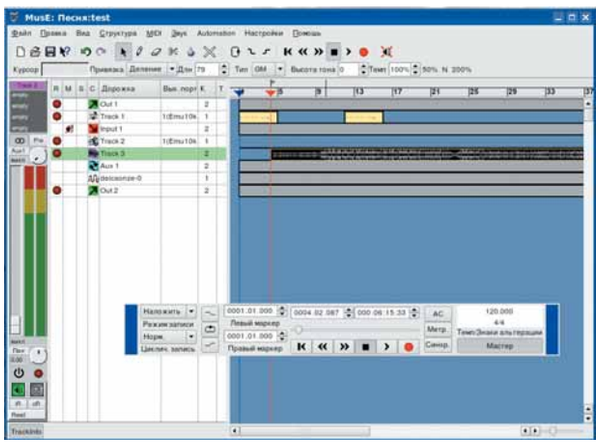
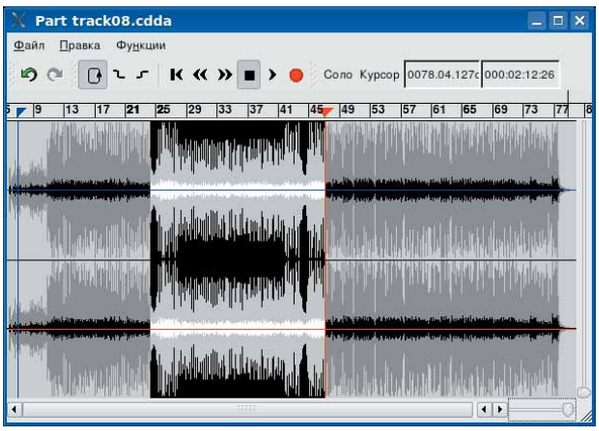


Рис. 6. Программный синтезатор *DeicsOnze*.



› **Рис. 7. Главное окно *MusE*.**



› **Рис. 9.**

Интерфейс *Muse* подобен *Rosegarden*, но с более очевидным уклоном в старый *Cubase* (**Рис. 7**).

Работа с MIDI

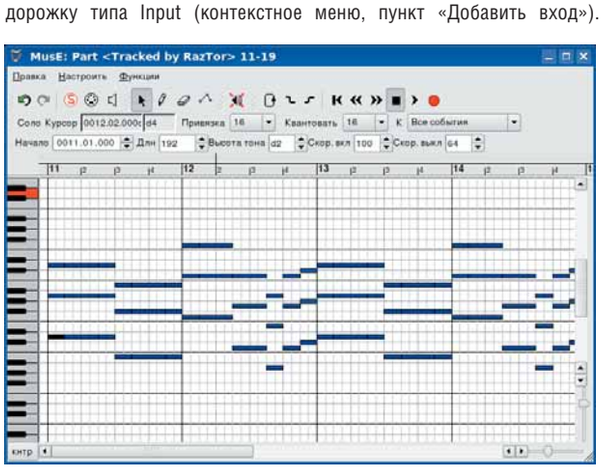
Muse предлагает для этих целей редактор ударных и пианоролл. Некогда был нотный редактор, но потом его убрали. Пианоролл более привычный, «кубэйсоподобный», нежели у *Rosegarden* – это касается и внешнего вида, и способа управления MIDI-контроллерами. Но есть одна деталь – в *Rosegarden*, когда вы рисуете в пианоролле ноту, она звучит. Дзынь! А в Mose почему-то не звучит. Зато другая мелочь – уже в пользу *Muse* – ноты могут отображаться разными цветами в зависимости от, допустим, громкости (**Рис. 8**).

Каждый виртуальный синтезатор имеет отдельную дорожку. На синтезатор можно навешивать до четырех LADSPA-эффектов, причем параметры эффектов полностью доступны для автоматизации. Автоматизация работает также в микшере.

В *Muse* есть два типа MIDI-дорожек – MIDI и «Дорожка с ударными». Отличие последней от первой в том, что рядом с названием отображается другая пиктограмма, да еще двойной щелчок мыши вызывает не пианоролл, а редактор ударных. Однако никто не мешает вам открыть в этом редакторе и обычную MIDI-дорожку. Особенность «Дорожки с ударными» – по умолчанию там выставлен MIDI-канал номер 1, но MIDI-устройство получает с этой дорожки данные на канал 10 (канал ударных). По мне, «Дорожка с ударными» лишняя, достаточно обычных MIDI-дорожек.

Работа с цифровым звуком

Сначала скажу об импорте сэмплов. Буду краток – работает, хотя форматов поддерживается всего два: WAV и некий загадочный Bin[Binary]. Теперь о записи звука. Чтобы осуществить ее, надо вначале создать дорожку типа Input (контекстное меню, пункт «Добавить вход»). Появится новая дорожка, где в свойствах приема сигнала (кнопка iR, input routing) надо выбрать входные порты (например, **alsa_pcm: capture_1**), а в выходном порту (кнопка oR) следует выбрать аудиодорожку, которая будет принимать сигнал.



› **Рис. 8. Пианоролл в *Muse*.**

Есть также дорожки эффектов, аналоги «кубэйсовским» FX-дорожкам. В *Muse* они называются Aux. Это пустые дорожки, не несущие никаких данных. Вы можете навешивать на Aux-дорожки эффекты, создавая таким образом как бы группы эффектов. Вы можете посылать сигнал на обработку в такую группу с обычной звуковой дорожки либо виртуального синтезатора. Это бывает полезно, если вы хотите обработать одинаковым набором эффектов несколько дорожек.

Muse позволяет открыть сэмпл в простеньком встроенном волновом редакторе (**Рис. 9**).

Функций в нем немного, все они касаются управления громкостью, затуханием (fade in/out и т.п.). Можно открыть сэмпл и во внешнем редакторе. Какой посоветовать? Был отличный редактор, ReZound, но его разработка остановлена. Еще мне нравится mhWaveEdit.

Прочие функции

В *Muse* хорошо развит механизм перенаправления (routing) сигнала. Можно создавать дорожки типа Out. Такие дорожки служат для вывода звука на реальный выходной порт (например, на выход звуковой карты или в JACK-клиент – тот же *Rosegarden*). Для Out-дорожки можно выбрать и входной порт, то есть одни дорожки можно посылать на одни выходы, другие дорожки – на другие выходы. Что это дает на практике? Если в компьютере установлено несколько звуковых карт или у звуковой карты есть несколько линейных выходов, то можно посылать дорожки на разные карты/выходы. Это вообще из области студийной звукорежиссуры.

Заключение

Сравнивая *Rosegarden* и *Muse*, нельзя сказать: вот эта программа лучше, а эта – хуже. Отчетливо видны плюсы и минусы обеих программ, а уровень развития тех или иных возможностей продукта определяет конечного пользователя. Так, если для вас важна работа с нотным станом, с верстской нотных листов, то *Rosegarden* будет, несомненно, наилучшим выбором. Однако для звукорежиссеров важна автоматизация, которую обеспечивает *Muse*. Поэтому если вы не просто делаете MIDI-аранжировку, а микшируете сложный проект, то *Muse* будет более полезен.

Обращая взгляд в мир Windows, нельзя не признать, что программ с такими возможностями и притом распространяемых свободно (я имею в виду оба критерия одновременно), для Windows просто не существует. Да, *Cubase*, *Nuendo* или *Sonar*, несомненно, мощнее, чем *Muse* и *Rosegarden*. Но подумайте и о ценах на коммерческое музыкально-звуковое ПО. Думаю, что *Muse* и *Rosegarden* более доступны большинству пользователей, чем программа за восемьсот долларов. LXF

Mandriva

Linux center
www.linuxcenter.ru

Mandriva Linux 2007
Дистрибутив Mandriva (ранее, до слияния с Connectiva называвшийся Mandrake), по праву считается идеальным для неподготовленного пользователя. Mandriva Linux невероятно прост в установке благодаря современному графическому установщику. Гордость дистрибутива – комплект графических утилит для настройки системы, объединенные в Mandriva Control Center. Установив Mandriva, вы сможете настроить абсолютно все, от звуковой карты до файерволла, не прибегая к помощи командной строки.

Cedega
Продукты Mandriva Linux 2007 Discovery и PowerPack включают TransGaming Cedega, позволяя запускать Windows-версии игр.

TransGaming Technologies создала уникальную технологию, позволяющую разработчикам собирать игры для одной системы и затем легко запускать их на других.
Технология TransGaming уникальна тем, что разработчикам не нужно думать о поддержке кроссплатформенности. TransGaming Cedega позволяет запускать популярнейшие игры, такие как Battlefield 2, Elder Scrolls IV: Oblivion, World of WarCraft, Civilization IV, Madden 2006 и множество других. Нет лучшего способа играть в Linux!

LinDVD
Продукты Mandriva Linux 2007 Discovery и PowerPack включают InterVideo LinDVD, позволяя просматривать DVD-диски абсолютно легально и без долгой настройки.
InterVideo -- лидирующий производитель программного обеспечения для просмотра DVD, а также мультимедиа-приложений.

Ia Ora
Последний раз Мандрива меняла тему рабочего стола ещё в версии 9.1, это была Galaxy. Для 2007 мы разработали с нуля новую тему назвав её Ia Ora ("здравствуйте" на языке франкоговорящих поленизийцев), поменяв там всё начиная от рамок окна заканчивая виджетами и иконками. Новая тема проще и легче, более яркая, хоть и используется всего несколько цветов.

3D
Поддержка нового трехмерного рабочего стола (AIGLX или Xgl) и новая утилита для его настройки (drak3d): Mandriva – единственный дистрибутив, предоставляющий легкий доступ к обеим технологиям, так что совместимость с оборудованием настолько широка, насколько это вообще возможно. Drak3d может сам определить лучший 3D-сервер. AIGLX и Xgl доступны в KDE и Gnome.

Новый RPMdrake
Mandriva разработала лучшую утилиту для установки, удаления и обновления программ. Новый RPMdrake дает вам информацию о том, какие пакеты у вас установлены, какие могут быть установлены, что рекомендует Mandriva, какие обновления доступны и т.д.

Cedega

Invictus Firewall
Mandriva разработала Invictus Firewall, высоконадежный брандмауэр для кластерных систем. Множество брандмауэров на машинах может быть настроено за считанные минуты, без отрыва их пользователей от работы.

CRM, ERP
Tiny ERP является бесплатным программным продуктом для интеграции данных и повышения продуктивности. Включен в Powerpack+. Пакет обрабатывает бизнес-процессы, продажи, финансы, поддерживает цепочки и менеджмент проекта... TinyERP дает возможность SOHO-пользователям наблюдать за их бизнесом.

Invictus Firewall

Joomla – открытая система управления контентом (CMS). Используется для всего, начиная с простых веб-сайтов и заканчивая сложными корпоративными приложениями. Мы выбрали Joomla потому, что она проста в установке, настройке и при этом надежна.

bitdefender
secure your every bit

BitDefender
BitDefender предоставляет решения для защиты сетей (антивирус, анти-спам, анти-spyware) для корпоративных и серверных нужд. BitDefender отслеживает трафик, блокируя вирусы, троянских коней и другие вредоносные приложения, защищая пользователей и их данные.

BitDefender – это:

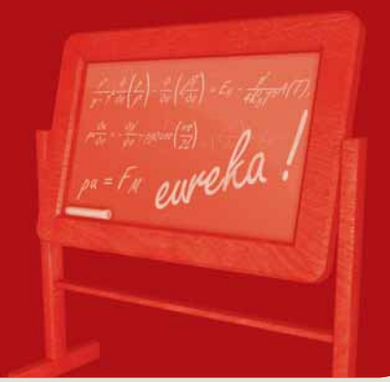
Комплексная технология анти-спама
Управление пользователями и группами
Совместимость с SNMP
Продвинутая система обновления

Закажите продукты Mandriva (Mandrakelinux) в Линуксцентре или у наших партнеров !
www.linuxcenter.ru/mandriva

Линуксцентр представляет авторизованные курсы Mandriva Linux, сертифицированные LPI.org
www.linuxcenter.ru/training

LINUX FORMAT

Учебники



Наши эксперты помогут вам с любым приложением Linux

ЕВГЕНИЙ БАЛДИН
Начинал с Агатов. Когда-то даже знал, что такое Робик.

Сначала привлечь, а затем обучать

«Вместе с влечением к новому и чужому это приводит к установке, которую можно назвать тягой к приключениям»

Конрад Лоренц. Обратная сторона зеркала

В последнее время активизировались обсуждения на тему, как имплантировать Linux в школьную среду.

На текущий момент подростки не знают что такое Linux и с чем его едят. Для осознания этого факта достаточно посмотреть на опрос проведённый на LOR (http://www.linux.org.ru) «Сколько вам лет?» На LOR нет школьников, а ведь казалось бы, это место просто создано для них.

В школе в большинстве случаев нет и не предвидится программ, нацеленных на обучение в среде GNU/Linux. У этой проблемы множество корней и основная из них та, что всем всё, как правило, «до фени». Поэтому, чтобы сдвинуть эту гору безразличия требуются титанические усилия или усилия многих, направленные примерно в одну сторону.

Прежде чем обучать, надо привлечь на свою сторону. Начать можно с малого, например, с красивых картинок. На текущий момент Linux Format достаточно пёстрый, чтобы привлечь подростка и понятный, чтобы не отпугнуть его. Это новая и неизведанная территория, которая сулит приключения. Принесите пачку уже номеров прочитанных в кабинет информатики и договоритесь, чтобы их могли брать все, кто захочет. Отдайте уже просмотренные DVD или сделайте новые. Пока просто принесите — это со временем, возможно, даст Вам сторонников и только тогда их можно начинать обучать.

E.M.Baldin@inp.nsk.su

В этом выпуске...

54 Как работать с RPM?

Fedora, SUSE, Mandriva – все они используют RPM для установки и обновления ПО. Если вы – новичок в Linux, разберитесь с ним вместе с **Энди Ченнелом**.

62 Уловка Beagle

Пол Хадсон расскажет, как использовать простейшие техники программирования и мощь Mono сотоварищи, чтобы всего за час собрать собственную поисковую систему.

58 Знакомство с SugarCRM

Запущенная CRM-система – неизменный атрибут успешно работающей компании. **Марк Бейн** начинает новую серию статей о достойной открытой альтернативе – SugarCRM.

66 Настройте свой загрузчик

Не стоит довольствоваться установками по умолчанию: **Ник Вейч** поможет выжать из *Grub* загрузку со своими заставками, с повышенным уровнем безопасности и кое-что еще...

70 Классика Glade

Визуальное программирование с GTK – это просто! **Андрей Боровский** научит вас создавать интерфейсы с помощью Glade 2.

74 И снова о консоли

Впечатлены возможностями dircolors? Узнайте, что еще умеет старый добрый терминал – читайте учебник Unix **Андрея Боровского**!

78 Java Server Pages

Неоднократно слышали, что дизайн надо отделять от бизнес-логики, но не знаете, с чего начать? **Александр Бабаев** познакомит вас с подходящей технологией!

82 Настройка PostgreSQL

Тюнинг сервера порой может быть увлекательнее тюнинга автомобиля – надо только знать, что подкрутить. **Евгений Балдин** готов прийти вам на помощь!

88 Классы LaTeX

Евгений Балдин классифицирует различные классы документов LaTeX. Иллюстрации прилагаются!

92 Живой мир Blender

В раскраске нуждается не только консоль! **Андрей Прахов** покажет, как вдохнуть жизнь в модели Blender, используя текстуры и анимацию.

Совет месяца: Боевая раскраска

Мы все стремимся сделать нашу жизнь ярче, и тот факт, что командная строка – родной интерфейс внутренностей вашей Linux-системы – текстовая, не означает, что вам придется мириться с монохромным терминалом. Сегодня мы покажем, как раскрасить его!

Добавить цвет можно различными способами и один из самых популярных – команда *dircolors*. Если вас смущает написание (Янки гоу хоум!), всегда можно создать символьную ссылку вида:

```
sudo ln -s /usr/bin/dircolors /usr/bin/dircolours
```

Dircolors использует цветовыделение для различных типов файлов, так что вывод простой команды **ls** превращается в настоящую радугу. Однако, вызванная сама по себе, dircolors просто возвращает список типов файлов и секретных кодов вроде **pi=40;33**: или ***.ogg=01;35:**. Первый элемент каждой записи – тип файла, за ним, после знака '=' следуют два числа, обозначающие цвет текста и цвет фона. Если вас смущают сокращения вроде **pi**, выполните **dircolors --print-database**. Вам объяснят, что **pi** – это просто символ конвейера (pipe), **40** – черный фон, а **33** – желтый цвет текста.

Если вы присмотритесь к выводу *dircolors* повнимательнее, то увидите, что он начинается с **LS_COLORS=** и заканчивается **export LS_COLORS**. Иными словами, dircolors просто устанавливает переменную окружения **LS_COLORS**. Вы можете добавить ее вывод к своему файлу **.bashrc**, чтобы устанавливать расцветку автоматически при входе в систему. В общем, запустите *dircolors* и вы посмотрите на консоль совсем другими глазами.

Да, имейте в виду: если вы по-прежнему не видите никаких цветов, вам может помочь команда **ls --color=auto**.

RPM: Пакеты

Благодаря RPM, управление пакетами в Fedora Core, SUSE и Mandriva происходит удивительно легко. Прогуляемся с **Энди Ченнелом** по процессу обновления и установки.

Наш эксперт

Энди Ченнел
Энди делает свои первые шаги в Linux уже шесть лет, а технологиями интересуется еще со времен Dragon 32.

Большинство пользователей и разработчиков Linux согласятся, что определяющее свойство операционной системы – наличие выбора. Так, у нас есть два (даже больше) рабочих стола, два загрузчика и две доминирующих системы управления пакетами: *APT* и *RPM*.

А вот определяющая черта пользователей и разработчиков Linux – пылкость, и если вы объявите, что «*APT* лучше всех», вас немедля атакуют. Однако до недавнего времени спорить с этим было трудно. На прошлом уроке мы убедились, что пришедшая из Debian Advanced Package Tool (*APT*) великолепна, и раньше пользователи дистрибутивов на основе Red Hat *Package Manager* (*RPM*) могли только завидовать своим друзьям с *APT*, запросто ставящим новые приложения.

Подобная ситуация не могла длиться долго; так и вышло. Сегодня *RPM* предлагает множество функций, позволяющих ей на равных состязаться с *APT*. На этом уроке я покажу, как дистрибутивы на основе *RPM* могут автоматически искать обновления пакетов и как использовать *RPM* для установки и удаления приложений. Вы сможете просматривать новые программы и мигом разрешать зависимости.

Что такое дистрибутив на основе *RPM*? *RPM* – это технология, вступившая в жизнь как часть дистрибутива Red Hat Linux. Она также нашла свое применение в других основанных на Red Hat дистрибутивах, включая Fedora Core, Mandriva и SUSE. То есть у трех самых популярных дистрибутивов Linux одинаковая система управления пакетами. *RPM*-пакеты, впрочем, обычно свои для каждого дистрибутива и версии программы, поэтому будет неразумно установить Fedora *RPM* на систему Mandriva, и не рекомендуется ставить SUSE 9.3 *RPM* на дистрибутив SUSE 10.2.

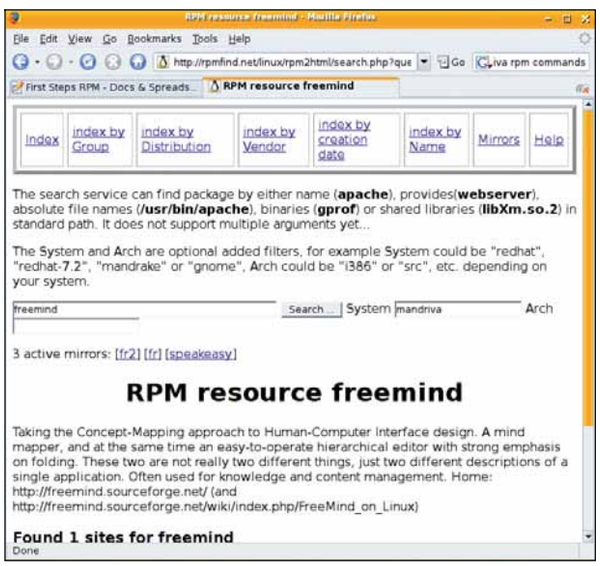
На прошлом уроке по *APT* я объяснял основные концепции управления пакетами; кто его не читал, может обратиться к врезке Глоссарий (*справа вверху*).

Обновляем дистрибутив

Как и в статье про *APT* в **LXF89**, мы начнем с рассмотрения автоматических средств, поддерживающих ваше ПО в актуальном состоянии, а потом займемся управлением пакетами через командную строку. Хотя вторая часть может показаться причудой, и к тому же негуманной для Первых шагов, это идеальный способ разобраться с *RPM*: большая часть программ управления пакетами в составе дистрибутивов на основе *RPM* – просто графические надстройки над командами. Зная эти команды, гораздо легче понять процесс установки программ с помощью *RPMDrake* в Mandriva, *Yast* в SUSE или *Package Manager* в Fedora.

Этот урок базируется на Mandriva Linux 2007, так как этот дистрибутив популярен среди новичков Linux, но большую часть описанных здесь процедур легко адаптировать и к двум другим *RPM*-дистрибутивам.

Итак, автоматические обновления. В Mandriva 2007 рабочим столом по умолчанию является KDE, иконка автоматических обновлений расположена в правом нижнем углу экрана рядом с часами (небольшой шарик). Щелчок левой кнопкой мыши открывает Менеджер Обновлений. Для этого вам понадобится пароль суперпользователя – как и для большинства задач установки и удаления ПО. Mandriva запросит некоторые данные, включая имя пользователя, пароль и адрес электронной почты (для подписки на сервис онлайн-обновлений). Вы получите допуск, а заодно зарегистрируете описание вашей



› RPM-пакеты доступны из многих источников. Проверьте, что берете пакеты именно для своего дистрибутива!

Месяц назад Управление пакетами в Ubuntu и других дистрибутивах на базе APT.

установить легко

машины (процессор, видеокарта и так далее). Если вы не хотите сообщать эту информацию, нажмите кнопку Cancel [Отмена], обновления получать все равно будет можно.

Проделав это, после запуска апплета обновлений нажмите на кнопку Check Updates [Проверить Обновления], чтобы узнать, нет ли радикальных заплаток или обновлений для приложений. Подумайте также, нужно ли вообще обновлять дистрибутив, особенно если вы сидите на модеме и вам не улыбается платить огромный телефонный счет за мелкую поправку к *OpenOffice.org*... Если хотите продолжить, нажмите Install Updates [Установить Обновления], и процесс обновления пойдет. Нижняя часть апплета обновления содержит много полезной информации, включая время последней проверки сервера, статус сети и оценку актуальности вашей системы.

По умолчанию этот процесс запустится автоматически при загрузке системы, но кое-кто (например, все те же пользователи модемов) не захочет, чтобы его компьютер каждый раз подключался к сети. Если вы один из них, то автоматический запуск легко отключить. Щелкните правой кнопкой мыши на иконке автоматического обновления в системном лотке и снимите галочку с опции Always Launch On Startup [Всегда запускать при загрузке]. Можно также вручную проверить обновления, выбрав пункт Check Updates [Проверить Обновления]: это пригодится и пользователям с высокоскоростным доступом, если они никогда не выключают свой компьютер.

Вот так получают автоматические обновления под Mandriva. Пользователи Fedora могут установить автоматические обновления через командную строку. Откройте терминал, наберите **su**, нажмите Enter, введите пароль root, затем наберите **chkconfig yum on**. Процедура для SUSE почти та же, что и в Mandriva, отличается только имя менеджера (*Yast* Online Update).

Простые команды

Поразмяв таким образом мозги, посмотрим, как пользоваться *RPM* из командной строки. Нужно будет скачать двоичный *RPM*-пакет с Интернета и работать от лица root. Для начала найдем подходящий пакет; одно из лучших мест для поиска – **http://rpmfind.net**. Здесь есть поисковая машина, ограничивающая результат по дистрибутивам и архитектуре. Запустите свой поиск (я собираюсь найти программу *FreeMind*) и затем найдите в выдаче результатов нужную версию. Слева на экране находится ссылка на описание пакета, а справа – ссылка на сам пакет для скачивания. Нажмите на нее, и файл скачается на ваш рабочий стол.

Теперь откройте терминал (помните, что он может называться *Console* в Gnome или *Konsole* в KDE и прячется где-то в меню Система) и наберите **su**, чтобы получить привилегии root. Скорее всего придется сменить каталог, так как *Firefox* и другие web-браузеры скачивают файлы на рабочий стол, а терминал почти всегда открывается в каталоге **/home/имя_пользователя**. Каталог сменить легко:

```
cd Desktop
```

cd как раз и значит ‘сменить каталог’, а **Desktop** – это рабочий стол, куда мы хотим перейти. Заметим заглавную букву D, так как Linux (в отличие от Windows) чувствителен к регистру. Теперь вы можете просмотреть содержимое каталога командой **ls**, или, если вы привыкли к DOS, **dir**. Используя **ls**, вы заметите, что файлы и каталоги отображаются разными цветами, и в каталоге **Desktop** вы должны увидеть свежескачанный *RPM*-файл. На рисунке внизу показан мой *RPM FreeMind*, выделенный красным цветом.

Глоссарий

» **Пакет** Двоичный файл, содержащий приложение. Пакеты доступны в различных форматах, включая APT и RPM (поэтому я и говорю «RPM-пакет»).

» **Менеджер пакетов** Приложение, управляющее установкой, обновлением и удалением пакетов.

» **Зависимости** Части программного обеспечения, которые требуются другим частям программного обеспечения, прежде чем последние могут быть установлены.

» **Обновление** Новая версия программы. Сюда могут входить новые функции, или решения вопросов безопасности, или исправления ошибок.

» **Двоичные файлы** Программное обеспечение в Linux часто поставляется как в виде исходных текстов, так и в двоичной форме. Двоичное приложение уже готово к установке, а исходные тексты необходимо компилировать.

» **Репозиторий** База данных по программам в сети (или на дисках), к которой может обращаться ваш дистрибутив.

Теперь мы именно там, откуда надо начинать установку. Вы заметите, что любое имя файла *RPM* состоит из имени приложения (‘*FreeMind*’ в моем случае), номера версии (0.8.0), номера релиза (6) и расширения *.RPM*. Может быть включено имя дистрибутива и/или версия, для которой предназначен пакет, а также архитектура, например, i386, PowerPC или ARM. Приложения, не привязанные к определенной архитектуре (как *FreeMind* в моем примере), содержат строку ‘noarch’ в имени файла.

Наконец, мы можем использовать простую *RPM*-команду, чтобы установить наше приложение:

```
rpm --install freemind-0.8.0-6.noarch.rpm
```

или

```
rpm -i freemind-0.8.8-6.noarch.rpm
```

Если вы хотите обновить существующее приложение на вашей системе, а не установить новое, то **i** в команде надо заменить на **U**.

Поддержка программного обеспечения

К сожалению, попытавшись установить *FreeMind* на Mandriva Free 2007 (бесплатная версия), вы наверняка наткнетесь на массу неудовлетворенных зависимостей.

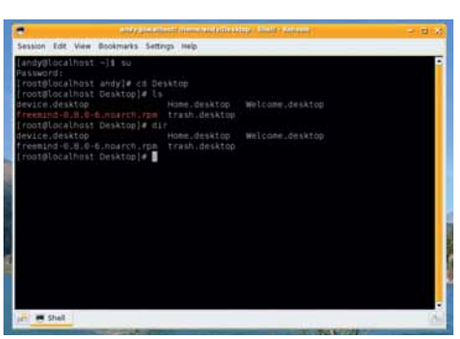
Если вам с ними не справиться, попробуйте *Urmpi*. Это инструмент командной строки, обращающийся к репозиториям программ для разрешения зависимостей при попытках установки приложений. *Urmpi* не имеет проблем с зависимостями, как *RPM*, поскольку сам ставит необходимые пакеты, однако кто-то должен позаботиться о создании пакета и поместить его в репозиторий.

Чтобы установить приложение с помощью *Urmpi*, начните так:

```
urpmi имя_приложения
```

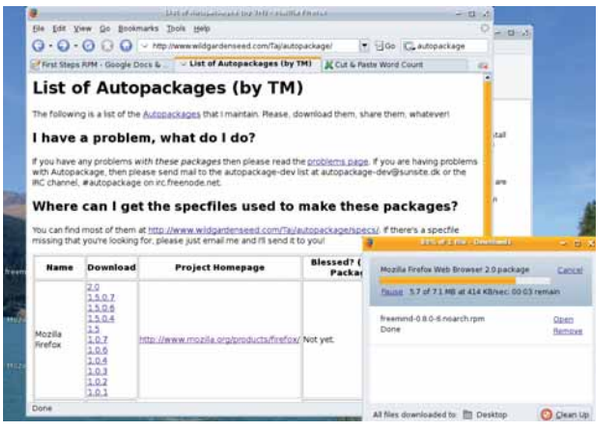
Номер версии вам не требуется, так как программа установится с самой последней версией.

К Mandriva уже прилагается ряд предустановленных *Urmpi*-репозиториев: это CD или DVD, на которых поставляется дистрибутив, а также любой источник обновлений, настроенный при начальной установке дистрибутива. Но можно добавить и другие репозитории, если в предустановленных »



› Просмотр файловой системы Linux с помощью командной строки. Наш скачанный пакет выделен красным цветом.

Шаг за шагом: Автопакеты для установки

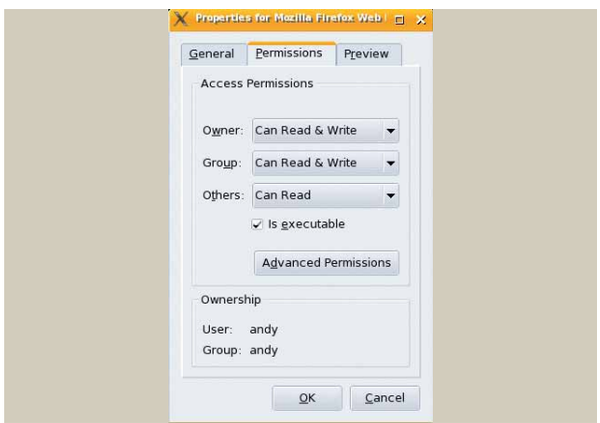


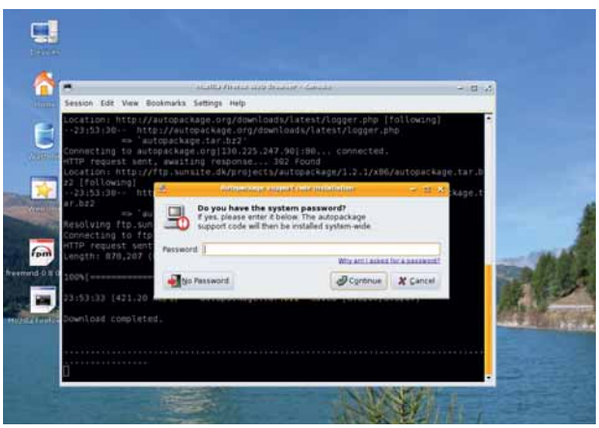
1 Познакомьтесь

Autopackage – альтернативный способ работы с пакетами. Для начала, посетите **www.autopackage.org**, скачайте последнюю версию *Autopackage* и найдите ссылки на программы, для которых имеется Автопакет.

2 Скачайте Autopackage

Загрузите Автопакет, например, *Inkscape*, с **www.autopackage.org** на ваш рабочий стол. Далее, щелкните правой кнопкой мыши на файле и выберите Properties [Свойства].



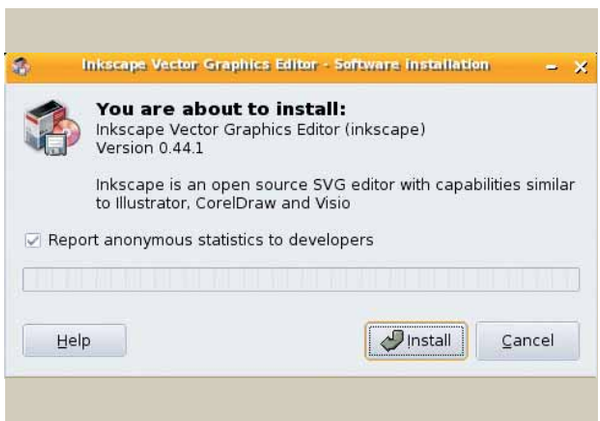


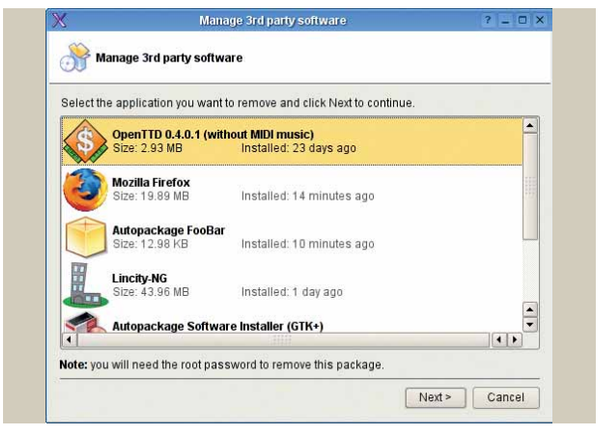
3 Сделайте его исполняемым

В диалоге Properties выберите вкладку Permissions [Права доступа] и нажмите на Executable [Исполняемый] (в KDE) или Allow Executing File As Program [Позволить выполнять файл как программу] (в Gnome). Это позволит Linux запускать скачанный файл как приложение. Нажмите OK.

4 Очистите стол, затем установите

Удалите старую версию вашего приложения, если оно установлено. Теперь дважды щелкните на скачанном файле и следуйте подсказкам. Перед установкой самого приложения произойдет скачивание и установка всех требуемых пакетов.





5 Запустите новое приложение

Теперь приложение появится в нужном месте вашего меню, готовое к использованию. В следующий раз, когда вы скачаете Автопакет из сети, вам автоматически предложат его установить.

6 Удаление Autopackage

Чтобы удалить Autopackage, наберите **package remove имя_приложения** в командной строке. Или используйте графические инструмент в меню System [Система] > Manage 3rd Party Software [Управление сторонними программами]. Просто установите радиокнопку рядом с именем приложения.

требуемого приложения нет – а для поиска нужного репозитория имеется великолепный сайт. Чтобы найти и добавить новый репозиторий, зайдите на **http://easyurmpi.zrb.orf**, введите номер версии Mandriva, архитектуру и менеджер пакетов, и следуйте подсказкам. На последнем этапе появится список команд, которые необходимо набрать (их можно скопировать и вставить) в командной строке от лица root. В зависимости от количества добавленных источников и скорости вашего интернет-соединения эта процедура может занять от нескольких минут до получаса. Например, добавление источника Penguin Liberation Front (он предоставляет множество несвободных приложений, особенно полезных для проигрывания мультимедиа) на мой компьютер потребовало скачать 50 МБ. Код, генерируемый EasyUrmpi, имеет следующий формат:

```
urpmi.addmedia plf-free ftp://ftp.univ-orleans.fr/logiciel-libre/plf/
mandriva/2007.0/free/release/binary/i586/ with hdlist.cz
```

Он содержит команду (**urmpi.addmedia**), сопровождаемую описанием репозитория и указанием на расположение репозитория и списка пакетов (часть **hdlist.gz**). Поэтому, например, добавив все предлагаемые EasyUrmpi источники, вы сможете набрать в консоли **urmpi flashplayer-plugin** и установить плейер от Adobe в *Firefox*, не задумываясь о XPI, tar.gz и прочих файлах.

В *URPMi* есть полезные команды, позволяющие выяснить, содержится ли искомое приложение в ваших репозиториях. Это: **urmpq имя_приложения**, она осуществляет простой поиск; **urmpq -i** произведет поиск и отобразит информацию о каждом найденном пакете; и **urmpq --fuzzy**, отображающая список приложений, напоминающих то, что вы ищете – например, поиск '*Firefox*' выдаст все пакеты, в которых упоминается данное слово.

Наконец, с помощью *Urmpi* можно обновить все приложения на вашей системе – как мы уже делали автоматически – до самых последних версий. Команда для этого будет такой:

```
urpmi --auto-select
```

Если нужно удалить программное обеспечение с вашей системы, сгодится старый добрый *RPM*. Не обязательно помнить все детали (например, номер версии) удаляемого пакета. Команда **rpm -e имя_приложения** позаботится обо всем.

Графическая красота

Я хотел показать вам, как пользоваться командной строкой, потому что хоть дистрибутивы и могут иметь свои собственные графические инструменты для управления программным обеспечением, все они основаны на тех же командах *RPM*-системы. Поэтому, овладев *RPM* из командной строки в Mandriva, вы сможете давать советы своим друзьям с другими дистрибутивами, и выглядеть элитным Linux-хакером. А сейчас рассмотрим альтернативу *Urmpi* и терминалу: скачаем пакеты с помощью графического инструмента Mandriva.

Сначала выберем в стартовом меню System [Система] > Configuration [Настройка] > Packaging [Пакеты] > Install, Remove And Update Software [Установить, Удалить и Обновить Программы]. Появится новое окно (точнее, часть Центра Управления Mandriva Control Center), содержащее четыре опции. Если вы выберете четвертую – под названием Select From Where Software Packages Are Downloaded When Updating The System [Выбрать источники для скачивания пакетов при обновлении системы] – то появится другой экран, содержащий все репозитории, добавленные ранее с помощью *Urmpi*. Вы можете включить или выключить их простым щелчком мыши. (Если вы не добавляли дополнительные репозитории, сделайте это сейчас: при использовании EasyUrmpi, как я показывал, вам будет выдан набор команд, которые можно просто скопировать и вставить в консоль.)

Две самых интересных штуки на этом экране – кнопки установки и удаления, расположенные вверху. Выберите первую. Ее окно содержит три панели и панель поиска наверху (см. картинку ниже). Слева представлены категории приложений, например, Игры и Утилиты, а справа – соответствующие приложения (вверху); если выбрать какое-то из них, внизу появится информация о нем. Можно, например, выбрать X11 > *Compiz* и узнать поподробнее о менеджере *Compiz*. Если вы установите радиокнопку рядом с доступными *Compiz* пакетами, менеджер пакетов выберет все пакеты, от которых он зависит.

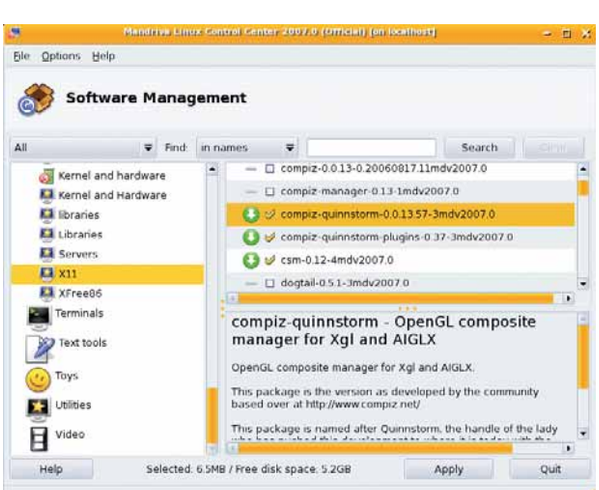
Поздравляем, вы гуру...

После нажатия кнопки Apply [Применить] пакеты будут скачаны и установлены с удаленных репозиториев или Mandriva DVD. Установив приложение, нажмите кнопку Close [Закрыть] для возврата в окно управления пакетами. Список установленных пакетов можно вызвать при помощи инструмента удаления, а заодно и ликвидировать ненужные пакеты.

Со знанием *RPM*-команд, графические инструменты Mandriva должны стать вам понятны: в общем, они дают простейший способ установить или удалить программу. Доступные репозитории содержат большую часть программ, необходимых для повседневной работы, а если нет, то EasyUrmpi позволит их добавить, и вам не придется снова перерывать всю сеть в поисках нужного приложения. А если придется, то сначала попробуйте установку с помощью *Urmpi*, а *RPM* употребляйте как последнее средство. Надеюсь, я показал вам, что командной строки бояться не надо, и что для некоторых задач (например, для добавления репозиториев в *Urmpi*) она быстрее, чем графические инструменты. Ваше путешествие по Linux продолжается... LXF

Скорая помощь

Если вы должны использовать RPM-пакеты, полученные не из вашего репозитория, то убедитесь хотя бы, что они подходят к вашему дистрибутиву и версии, в противном случае поломка будет обеспечена. Вы можете протестировать, правильно ли установится RPM, набрав **rpm --test -i**.



Графический подход Mandriva к установке ПО. Видите, я выбрал три пакета для установки. Готовы? Жмите Apply [Применить].

Через месяц Используйте права доступа, чтобы никто не лез в ваши данные.

SugarCRM: Как

ЧАСТЬ 1: Вашему бизнесу не повредит автоматизация работы с заказчиками. **Марк Бэйн** показывает, как приспособить для этих целей лучшее открытое ПО.

Наш эксперт

Марк Бэйн
Свою первую программу Марк написал еще для ZX 81. Он уже много лет работает с Unix и Oracle, а кроме того, читает лекции и пишет.

Добро пожаловать в новую серию! На первом уроке я покажу, как установить отличную CRM с открытым исходным кодом. А что такое CRM? Это Customer Relationship Management – системы управления взаимодействием с клиентами. Поняли? Для тех, кто не понял, Майкл Уайтхед-младший [Michael JR Whitehead] в книге "*Implementing SugarCRM*" определяет эти системы так:

> «CRM предназначены для хранения и обработки информации о ваших клиентах – не только о продажах, но и о дальнейших взаимоотношениях и поддержке. Системы обязаны давать хотя бы минимум информации о компаниях, с которыми вы работаете и о том персонале, с которым вы в них сотрудничаете».

Между прочим, у вас почти наверняка все это есть. Мало того, что вы храните свои контакты в адресных книгах *Thunderbird*, *KMail* или (содрогаюсь) *Microsoft Outlook*, а счета в *OpenOffice.org Calc* или *Microsoft Excel* – небось еще и создали аж базы данных *Base* или *Access*, и записываете туда все свои сделки. Все как будто прекрасно. Но дело в том, что все ваши коллеги проделывают те же операции со своим почтовым клиентом, электронными таблицами или базой данных. Вроде ничего плохого, разве что данные многократно дублируются – и это бы не беда... пока не окажется, что данные не стыкуются.

Представим себе следующую ситуацию. Пусть в число ваших заказчиков входит фирма Linux PCs Ltd. Вы записали ее почтовый адрес: Penguin House, Southwaite CA4 0AK. Но Фред справа уверяет, что правильный адрес – Penguin House, Carlisle CA4 0AK, а у Мэри слева он значится как Penguin House, Southwaite, Carlisle. Кто же прав? Настоящий адрес знает один Генри: фирма переехала, а он забыл всем про это сообщить.

Разрулить весь бизнес

Значит, CRM спасет вас от подобных ситуаций? Не только! Возможности CRM гораздо шире. Вот что вам предоставляется (опять цитирую Уайтхеда):

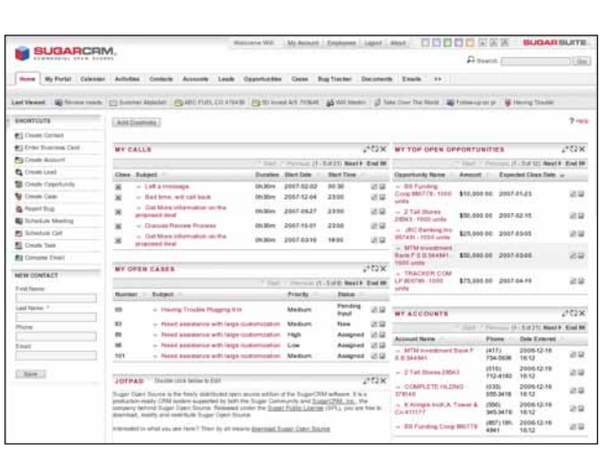
- автоматизация работы персонала;
- мониторинг потенциальных возможностей;
- мониторинг канала продаж;
- определение групп сотрудников и территорий их деятельности;
- анализ основных источников прибыли;
- работа с каталогом продукции;
- создание выписок для клиентов;
- гибкая система формирования отчетов;
- мониторинг оказываемой клиентам технической поддержки;
- разработка план-графика выполняемых работ;
- хранение контактной информации организации;
- унификация взаимодействия с системой для различных пользователей;
- система контроля версий для документов.

Хорошо, а какая с этого корысть? Во-первых, уменьшатся ваши издержки. Информация будет вводиться однократно, что исключает разночтения внутри вашего предприятия. Во-вторых, ваши сотрудники всегда будут в курсе событий на предприятии, и почувствуют себя полноправными их участниками, а не винтиками (это касается и персонала, и руководителей). Заказчики также будут довольны, имея дело с компетентными, хорошо осведомленными партнерами. Как следствие, возрастут продажи.

Вы справедливо спросите: а кто я такой, чтобы об этом писать? Основное мое занятие – реализация и адаптация Clarify CRM; если у вас есть знакомые в фирме Vodafone, спросите их о проекте Cascade. Могут ли открытые CRM-системы достигать таких же результатов? Естественно, могут!

Требования к серверу

У *SugarCRM* нет отдельного пользовательского интерфейса – все дела-



› *SugarCRM* управляется с номерами телефонов, номерами счетов и многим другим! Вам не по душе внешний вид? Его можно поменять.

его установить

ется через web-браузер. Это обеспечивает независимость системы от платформы. Полагаю, вы предпочитаете Linux – так что радуйтесь: мы займемся платформой LAMP (Linux-*Apache*-*MySQL*-PHP). (Намек поняли? Именно это и надо установить на вашем сервере…)

Начнем с сервера. Я установил *SugarCRM* под Debian на стареньком компьютере Patriot, и мне этого хватает. Но для предприятий рекомендую следующие спецификации:

До 10 пользователей:

- Жесткий диск IDE или SCSI емкостью 80–100 ГБ;
- 1 ГБ оперативной памяти с ECC (обнаружение и исправление ошибок);
- Процессор Athlon, Pentium 4, Xeon или Opteron с тактовой частотой не менее 2 ГГц.

От 10 до 100 пользователей:

- Жесткий диск SCSI емкостью 150 ГБ;
- 2–3 ГБ оперативной памяти с ECC;
- Процессор Xeon или Opteron с тактовой частотой не менее 2 ГГц.

Более 100 пользователей:

- Жесткий диск SCSI емкостью 300 ГБ;
- 4–8 ГБ оперативной памяти с ECC;
- Два процессора Xeon или Opteron с тактовой частотой не менее 2 ГГц.

Все тот же Уайтхед, занимавшийся *SugarCRM* с 1999 г., также советует использовать:

- Сеть Ethernet со скоростью 1 Гб;
- Источник бесперебойного питания (UPS) – бесценная штука, особенно если учесть, как они сейчас дешевеют;
- SUSE Linux. Хотя лично я предпочитаю использовать на моих серверах Debian (SUSE 10.1 у меня тоже стоит, на рабочей станции).

Подготовка

Linux, понятное дело, у вас есть; осталось добавить *Apache 2*, *MySQL* и PHP.

Apache 2 устанавливается легко. В дистрибутиве Debian для этого нужно выполнить следующую команду

```
sudo apt-get install apache
```

После установки убедитесь, что в конфигурационном файле web-сервера включена обработка PHP-скриптов. Впрочем, в Debian это делается по умолчанию.

Установка *MySQL* ничуть не сложнее:

```
sudo apt-get install mysql-server
```

Все необходимые таблицы в базе данных *SugarCRM* позже создаст сама, а сейчас вам нужно лишь установить пароль для пользователя root. Подключитесь к базе и выполните следующий SQL-запрос:

```
update user
set password=password('ваш_пароль')
where user = 'root';
```

Пароль не забывайте: он потребуется при установке.

Угадайте-ка, трудно ли установить PHP? Вот именно, нетрудно: процедура все та же, sudo **APT-get install**, только на сей раз надо приписать **php**. Я уже упоминал, что файлы конфигурации *Apache* надо правильно настроить на обработку PHP-скриптов, но в Debian уже все сделано. Однако PHP все-таки придется настраивать. Поменяем системные переменные PHP в файле **php.ini**, установив:

- **MEMORY_LIMIT в значение 64M**. *SugarCRM* во время установки читает его неправильно и остается при значении по умолчанию: 8 МБ. Если не сменить эту переменную вручную, *SugarCRM* скорее всего упрется в лимит памяти и оставит вас с пустым экраном.
- **MAX_EXECUTION_TIME в значение 90**. Так мы предотвратим появление таймаутов при импорте больших объемов данных.
- **MAX_INPUT_TIME в значение 300, а POST_MAX_SIZE в значение 25M**. И то, и другое поможет загружать без ошибок файлы большого размера.
- **Увеличенное значение SESSION.MAXLIFETIME**. В течение этого времени *SugarCRM* ожидает ваших действий, а потом отключается (по умолчанию 1440 сек = 24 мин). Лучше взять его побольше, например, 3600 секунд (1 час). Нет ничего хуже, чем намучиться, составляя длинное почтовое сообщение, и тут же убедиться: приложение успело отсоединиться, и отправить почту нельзя.

Теперь все готово для установки *SugarCRM*.

Позаботимся о сотрудниках

Итак, основная и самая трудная часть работы позади (а она показалась вам трудной?). Теперь загрузим архив программы с сайта www.sugarcrm.com, распакуем его в локальный каталог и запустим скрипт установки в браузере. Осталось только следовать инструкциям, появляющимся на экране.

Рассмотрим эти этапы поподробнее. Начнем со структуры каталогов *SugarCRM*. После распаковки архива вы обнаружите новый каталог, с именем вроде **SugarOS-Full-4.5.0g**. Этот каталог нужно скопировать в корневую директорию web-сервера (где-то в **htdocs**). Для доступа к *SugarCRM* из вашего браузера в адресной строке наберите 'http://<имя_web-сервера>/SugarOS-Full-4.5.0g'. Например, у меня путь был таким: 'http://hector/SugarOS-Full-4.5.0g' (имя моего сервера – Hector).

Однако это не самый удобный вариант URL. Сотрудники вашей фирмы не знают, что такое *SugarCRM* (им на это наплевать) – они просто хотят максимально облегчить свою работу. Пусть имя каталога говорит само за себя.

Например, вы владелец издательства и решили выпускать новый журнал – назовем его хотя бы *Linux Format* (правда, красиво?). Тогда переименуем каталог в **lxf**. Пользователи будут набирать URL 'http://hector/lxf'. Коротко и удобно, а действия *SugarCRM* никак не затрагивает. Каталог можно переименовать и попозже, но пользователей это в восторг уже не приведет. Лучше зайти в корневой каталог сервера и создать несколько символических ссылок на него, введя, допустим,

```
ln -s lxf linuxformat
ln -s lxf magazine
```

Доступ к программе даст любой из адресов 'http://hector/lxf', 'http://hector/linuxformat' или 'http://hector/magazine'.

Мы почти готовы установить *SugarCRM*, осталось только определить, кто по умолчанию будет владеть файлами *SugarCRM:* это сильно повлияет на остаток процесса. Выясните, кто владелец вашего web-сервера по умолчанию, и назначьте его же владельцем каталога lxf. Например, в моем дистрибутиве Debian это 'www-data'. Вообще-то я предусмотрел, что все файлы разрешают запись из группы, и добавил в группу себя, а сервер *SugarCRM* установил на моей рабочей станции: так я могу вносить все изменения прямо в файлы конфигурации.

Наконец-то установка

Вы, небось, уже открыли браузер и набрали URL вашего *SugarCRM*. При правильной настройке вашего сервера *Apache* на поддержку »

Скорая помощь

До начала установки SugarCRM задайте вручную значение системной переменной MEMORY_LIMIT в файле php.ini равным 64 МБ. Иначе программа будет использовать значение, по умолчанию равное 8 МБ, и процесс завершится аварийно из-за нехватки памяти.

» скриптов PHP вы видите стартовое окно программы установки; если не видите, значит, нужно отредактировать файлы конфигурации *Apache* и перезапустить сервер. Пора жать кнопку “Start”.

» Шаг 1

На первом этапе установки перед вами окошко с лицензией продукта. Прочитав ее и установив соответствующую галочку, нажмите Next; вы попадете на самый полезный экран во всем процессе.

» Шаг 2

Второй шаг – системная проверка. При этом выводится список всех компонентов, от которых зависит *SugarCRM*, с указанием их статуса (существует в приемлемой версии; не существует, но без него можно обойтись; не существует, но обязателен). Вам придется установить отсутствующие компоненты и обновить устаревшие.

Обычно не хватает модулей интерпретатора PHP. Что делать дальше, зависит от дистрибутива. В Debian недостающие модули легко установить через утилиту **apt-get**. Раздобыв все необходимые компоненты, нажмите кнопку Re-Check для повторной проверки.

И помните, нельзя верить строке, извещающей о лимите оперативной памяти в PHP: даже если вы установили этот предел как 64 МБ, вы прочтете:

‘PHP Memory Limit >= 8M OK’.

» Шаг 3

Следующая задача – настройка базы данных *MySQL*. Фактически, настраивать ничего не придется – нужно лишь указать программе установки следующие параметры:

» **Host Name Имя хоста**. По умолчанию ‘localhost’, и менять его незачем.

» **DataBase Name Имя базы данных**. По умолчанию ‘*SugarCRM*’. Лучше заменить чем-то более осмысленным, тем более, что переименование позволит иметь на сервере больше одной системы *SugarCRM*. Не забудьте установить флаг Create DataBase (Создать базу данных).

» **Username/Password For SugarCRM Имя пользователя и пароль для SugarCRM**; они необходимы для выполнения запросов к базе данных программы. По умолчанию ‘*SugarCRM*’, но опять же лучше заменить его более подходящим. Не забудьте отметить флаг Create User (Создать пользователя).

» **Populate DataBase With Demo Data Занести в БД данные демонстрационного примера**. Хотите – заносите и потренируйтесь с ними, хотите – нет. Лично мне это ни к чему.

» **Privileged DataBase User Name Привилегированный пользователь БД**. Обычно это ‘root’, но если вы создали другого пользователя с необходимыми привилегиями, можете использовать и его. Учтите, он должен иметь право управлять базой данных и создавать пользователей и таблицы (привилегии ‘data*Base*’, ‘create user’ и ‘create tables’).

» **Privileged DataBase User Password Пароль привилегированного пользователя**. Вот почему мы устанавливали пароль для ‘root’ при первом запуске *MySQL* (помимо заботы о безопасности, разумеется).

Все необходимые параметры указаны – нажимаем на кнопку Next. База данных будет создана позже, и если вы что-либо передумаете, можете вернуться к предыдущим шагам установки и внести изменения.

» Шаг 4

Произведем базовую настройку программы. Она включает установку URL-адреса для доступа к программе и задание пароля для администратора admin. URL на данном этапе не особо важен, и потом его можно будет поменять. Оставим тот, что по умолчанию. А вот пароль для admin понадобится при настройке системы и добавлении пользователей, так что запомните его хорошенько.

Автоматическое обновление программы (флаг Enable Sugar Updates) может производить только администратор системы, и лучше этот флаг не устанавливать: у всех остальных пользователей он вызовет сообщение об ошибке. Флаг Advanced Site Security (дополнительные опции безопасности) то ли включает, то ли выключает их – неясно; поэтому я оставил его в состоянии Use Defaults (настройка по умолчанию).

Как и раньше, нажатие кнопки Next не вносит необратимых изменений, и при желании вы сможете все переиграть.

» Шаг 5

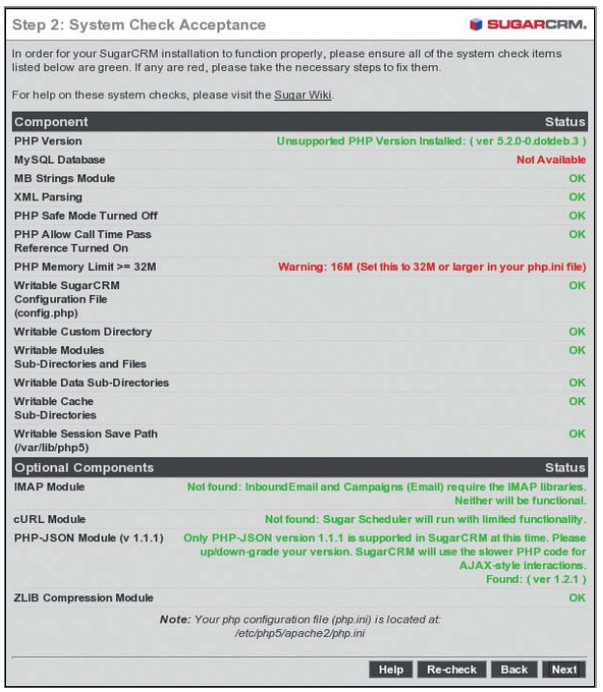
Предпоследний этап. Делать ничего не надо, только проверить на экране, все ли данные указаны верно. Это ваш последний шанс изменить настройки. Готовы? Жмите кнопку Next.

Скорая помощь

Придумайте осмысленное название для каталога, в который установлен *SugarCRM*. Это существенно облегчит работу с системой вашим сотрудникам.

› Увидав стартовое окно программы установки SugarCRM, считайте, что все почти готово.

» **Что читать** Книга Майкла Уайтхеда издана в Packt Publishing.



Step 2: System Check Acceptance — SUGARCRM.

In order for your SugarCRM installation to function properly, please ensure all of the system check items listed below are green. If any are red, please take the necessary steps to fix them.

For help on these system checks, please visit the Sugar Wiki.

Component	Status
PHP Version	Unsupported PHP Version Installed: (ver 5.2.0-0.dotdeb.3)
MySQL Database	Not Available
MB Strings Module	OK
XML Parsing	OK
PHP Safe Mode Turned Off	OK
PHP Allow Call Time Pass Reference Turned On	OK
PHP Memory Limit >= 32M	Warning: 16M (Set this to 32M or larger in your php.ini file)
Writable SugarCRM Configuration File (config.php)	OK
Writable Custom Directory	OK
Writable Modules Sub-Directories and Files	OK
Writable Data Sub-Directories	OK
Writable Cache Sub-Directories	OK
Writable Session Save Path (/var/lib/php5)	OK
Optional Components	**Status**
IMAP Module	Not found: InboundEmail and Campaigns (Email) require the IMAP libraries. Neither will be functional.
cURL Module	Not found: Sugar Scheduler will run with limited functionality.
PHP-JSON Module (v 1.1.1)	Only PHP-JSON version 1.1.1 is supported in SugarCRM at this time. Please up/down-grade your version. SugarCRM will use the slower PHP code for AJAX-style interactions. Found: (ver 1.2.1)
ZLIB Compression Module	OK

Note: Your php configuration file (php.ini) is located at: /etc/php5/apache2/php.ini

Help | Re-check | Back | Next

› Шаг 2 – самый важный: нужно вовремя подметить все ошибки и исправить их.

»» Шаг 6

Сидите и наблюдайте за сборкой *SugarCRM* и созданием новой базы данных. Здесь бывают только 2 варианта ошибки. Первый – вы не установили пароль либо ввели его неправильно, и *SugarCRM* не может создать вам базу данных. Второй – неправильно указан объем памяти (*см. раздел Подготовка*). Придется повторить установку с самого начала, предварительно изменив строку ‘installer_locked =>true’ на ‘installer_locked => false’ в файле **config.php**, расположенном в каталоге программы.

Но вы-то следовали инструкциям, и я надеюсь, что никаких проблем у вас не возникло, а *SugarCRM* успешно установлен. В следующем месяце мы займемся тонкой настройкой системы, включая введение привычной вашим пользователям терминологии. Информация о *SugarCRM* и ее применении приведена на www.sugarcrm.com. Это не обязательно касается свободной версии, но все равно любопытно, как она используется в большом бизнесе. До следующего раза! LXF

Скорая помощь

Переменная среды SESSION.MAXLIFETIME определяет максимальное время бездействия пользователя, по истечении которого происходит автоматический выход из системы. По умолчанию это значение установлено в 24 минуты (1440 секунд).

Альтернативы SugarCRM

Существует большое количество платных программ CRM. Среди них *Microsoft CRM* (www.microsoft.com/smallbusiness/products/mbs/crm/detail.mspx), *Netsuite* (www.netsuite.com) и ведущее ПО для управления контактами, *SalesLogix* (www.saleslogix.com).

Почему же мы выбрали *SugarCRM*? Ответ прост: открытый исходный код. На данный момент более миллиона пользователей по всему миру используют *SugarCRM*. Разработку ведет SugarCRM Inc вместе с сообществом. Имеется большое количество расширений и дружелюбная поддержка на www.sugarcrm.com/forums. Если вы хотите установить такую систему у себя, бесплатная версия *SugarCRM* – лучший выбор.

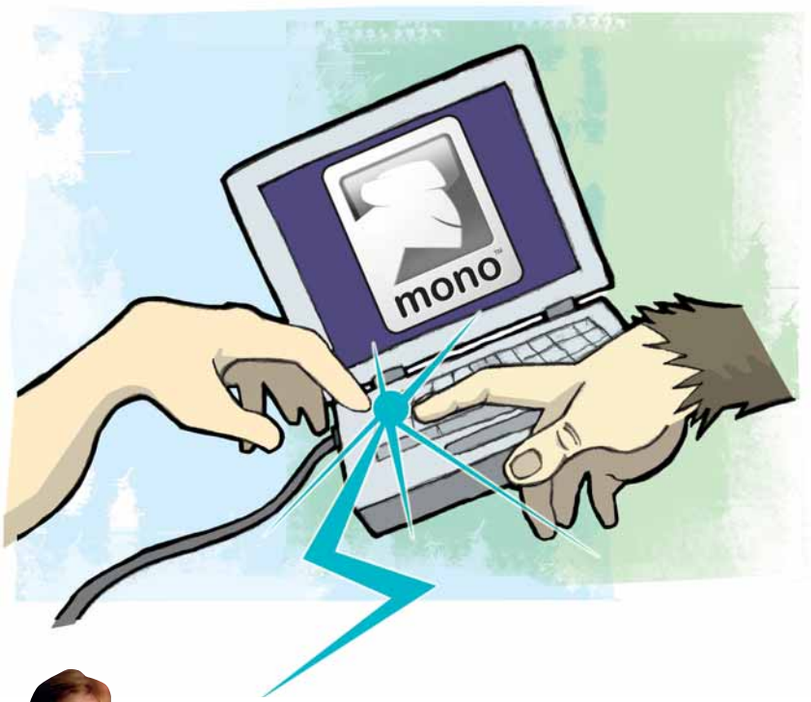
Доступны и платные версии программы, в том числе «арендуемые», т.е. пользователь платит за программу и за сервер. В платных версиях присутствуют дополнительные модули (например, прогноз продаж Sales Forecasting). Версия Professional обойдется вам в $40 за пользователя в месяц, а версия Enterprise, позволяющая работать с базой данных Oracle – в $75.

› Сравнение возможностей различных версий *SugarCRM*. www.sugarcrm.com/crm/products/

»» Через месяц Мы приспособим *SugarCRM* к нуждам ваших пользователей.

Mono: Связь с

Создать супер-поиск по файловой системе меньше чем за час? Наш штатный вождь обезьян **Пол Хадсон** делает невозможное не только возможным, но и простым...



Наш эксперт

Пол Хадсон полагает, что Mono – лучшая вещь со времен мультфильма Pinky and the Brain, и сейчас поддерживает два проекта на основе Mono на SourceForge.

Ваша миссия, если вы за нее беретесь, состоит в написании консольной программы, которая индексирует файловую систему пользователя в фоновом режиме, сканирует почту, RSS-ленты, историю браузера и содержимое файлов, а затем позволяет производить во всем этом молниеносный поиск. Уловили? OK, увидимся через час – удачи вам!

Да, мы собрались создать шедевр менее чем за 60 минут; по-вашему, без чуда не обойтись? Но прикиньте: какое небезызвестное Linux-приложение умеет осуществлять поиск по всем источникам данных? Думайте, думайте! Ясное дело, *Beagle*. А на какой платформе построен *Beagle*? Mono! А о чем этот учебник? О Mono! Итак, мы позаимствуем возможности у проекта *Beagle* – точнее, у его библиотек.

Что хорошо в .NET – это шанс встать на плечи гигантам, использовав их библиотеки. Конечно, C тоже такое допускает – например, программа *Ark RPG* имеет библиотеку *libarkrpg0c2a*, позволяющую другим программам на C вызывать ее функции, а не реализовывать их заново. Но такой подход не лишен проблем: например, *Gstreamer* разделен на несколько библиотек, и у каждой собственный номер версии. Рядовой пример – *libgstdataprotocol-0.10.so.0.8.1*. C не воспринимает новые версии библиотек без обновления самой программы, поэтому номера версий так важны – приложению подавай непременно *libgstdataprotocol-0.10*, а не 0.9 или 0.11.

Mono позволяет программистам просто потребовать «давай *Beagle*» и получить любую доступную версию. Такой подход не всегда срабатывает, особенно с некоторыми новыми Linux-приложениями, не достигшими версии 1.0. Причина, по которой у них такие маленькие номера версий (*Beagle* находится на стадии 0.2.14) – недостаточно стабильный программный интерфейс (application programming interface, API). API – это термин для обозначения имен функций, типов параметров и возвращаемых значений для данной библиотеки. Стабильность API означает, что если функция DoFoo() в версии 1.0 принимает целый аргумент и возвращает строку, то любая использующая ее программа будет работать и с версией библиотеки 1.01, потому что функция не меняется.

Самое главное в API – это 'I'-часть, собственно интерфейс (имя, параметры, возвращаемое значение): если он фиксирован, то неважно, как библиотека выполняет свою задачу, и пусть себе её версии меняются как угодно.

Резюмирую:

- Мы используем Mono, и *Beagle* тоже – значит, нам доступны преимущества *Beagle*.
- *Beagle* имеет API, предоставляющий в наше распоряжение его функциональность, но он не стабилен и в будущем может измениться (в прошлом он менялся довольно круто).
- Чтобы использовать *Beagle*, достаточно знать имена его функций и как к ним обратиться.

Вам это все еще кажется трудным? А так оно и есть. Но я ведь обещал научить вас всем навыкам, необходимым, чтобы к концу нашей серии уроков вы стали профессионалом, поэтому перейдем к делу...

Начнем с имени нашего приложения. Приложениям Gnome на основе Mono уже приелись имена с буквой G, и я решил выбрать что-нибудь смешное и продвинутое. Предлагаю *Poochy*, это имя малоизвестного песьеголового персонажа игры Nintendo (даже Майк про него не слыхал!) – оно вполне подходит к собачьей теме, заданной *Beagle* [англ. «гончий пес»].

Итак, запустите MonoDevelop, выберите File > New Project, затем

Новые проекты Mono начинаются с пустого окна в стиле *MonoDevelop* – это скоро станет второй натурой...

Месяц назад Мы построили суперчиталку лент новостей RSS.

библиотеками

C# > Console Project, уберите галочку с Create Separate Solution Subdirectory, затем введите имя '*Poochy*' в поле Name и нажмите кнопку New внизу. Вы получите старое доброе приложение Hello World – мы его 'программировали' в начале серии уроков; удалите строчки Console.WriteLine() и оставьте метод Main() пустым.

GTK и командная строка

Для этого проекта нам нужно использовать *Beagle* и – не удивляйтесь – *GTK*, поэтому нажмите правой кнопкой мыши на References на панели слева (сразу над Resources, AssemblyInfo.cs и Main.cs) и выберите Edit References. В появившемся окне переключитесь на вкладку Packages, где выберите *Beagle* (версия 0.0.0.0 – я предупреждал, что он далеко не стабилен, не так ли?) и *GTK*-sharp (версия 2.10.0.0). Эти номера просто сообщают, какие версии установлены в данный момент – если у кого-то другая версия *Beagle*, Mono преспокойно обойдется той, что есть.

Тем самым мы запросили для использования библиотеки *Beagle* и *GTK#*, но чтобы действительно их использовать, необходимо прописать операторы using в начале исходного кода. По умолчанию, единственная строка using –

```
using System;
```

Библиотека System включает основные функции для написания консольного приложения, включая доступ к самой консоли для обеспечения ввода-вывода. Мы хотим включить в этот список *Beagle* и *GTK*, так что добавим следующие две строки:

```
using Beagle;
using Gtk;
```

В методе Main() необходимо вызвать *Beagle* и сказать ему, что искать. Это можно сделать через Query, специальный объект *Beagle*. Этот объект также умеет запускать наши собственные методы при наступлении определенных событий – например, найдя что-нибудь, отвечающее поисковому запросу. Вот как все это выглядит в C# – поместите этот код туда, где находился Console.WriteLine():

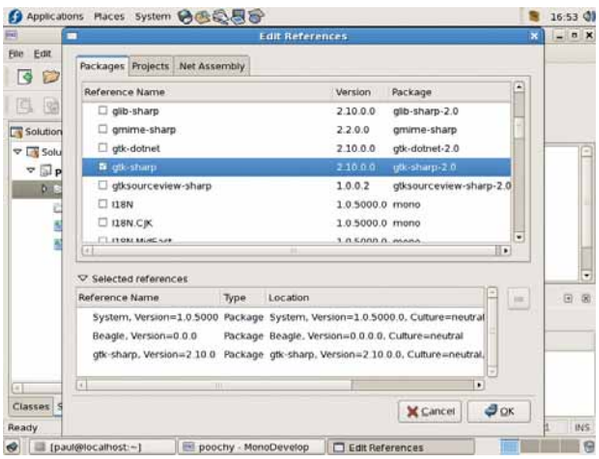
```
// Создаем новый запрос и добавляем два метода обработчика событий
Query q = new Query();
q.HitsAddedEvent += OnHitsAdded;
q.FinishedEvent += OnFinished;
// Сообщаем Beagle, что запрос содержится в первом параметре командной строки
q.AddText(args[0]);
// Начинаем поиск
q.SendAsync();
```

В общем, несложно, но это еще не вся сказка: найдя файл, RSS-ленту или что там еще совпадет с нашим запросом, *Beagle* всегда вызывает метод OnHitsAdded(), а закончив поиск, вызовет метод OnFinished() – чтобы код скомпилировался, необходимо реализовать оба этих метода. Для успешной компиляции достаточно вставить два пустых метода:

```
static void OnHitsAdded (HitsAddedResponse response) { }
static void OnFinished(FinishedResponse response) { }
```

Объекты HitsAddedResponse и FinishedResponse вообще-то дают массу полезной информации, но пока мы их проигнорируем. Продолжим: нажмем F8, чтоб скомпилировать первый релиз *Poochy* (храбро назовем его 0.1), затем откроем окно терминала и перейдем в **/каталог/где/находится/poochy/bin/Debug**. Пора протестировать нашу ищейку.

Poochy необходимо запускать из командной строки, потому что



› Как воспользоваться мощью GTK? Поставьте галочку рядом с этим именем в References, затем запомните ее API.

он читает arg[0], который мы должны передать из командной строки. Запустите следующую команду:

```
mono poochy.exe foo
```

Эта команда заставит *Beagle* искать в своем кэше аргумент 'foo', и для выполнения запроса понадобится около секунды (замените foo на что-нибудь заведомо присутствующее в вашем кэше *Beagle*). Но тут вы обнаружите, что ничего не выводится – ни файлов, ни чего-либо еще, содержащего 'foo'. Чья это проблема – *Beagle* или наша? *Beagle* вне подозрений, значит, проблема в нашем коде. Расследуем...

Гонка на время

Наша программа не работает по двум причинам. Более заметная из них – пуст метод OnHitsAdded(), вызываемый *Beagle* при нахождении совпадения, в который *Poochy* должен вывести результат. Поэтому Mono безмолвствует, даже и получив что-то от *Beagle*. Проще всего заставить *Poochy* выводить URL каждого найденного *Beagle* объекта, например, так:

```
foreach(Hit hit in response.Hits)
{
Console.WriteLine("Hit: " + hit.Uri);
}
```

Если вы запустите программу, то увидите, что она опять ничего не выдает, хотя *Beagle* должен бы найти то, что вы просили. Проблема в том, что наш поиск использует метод **SendAsync()**, то есть *Beagle* выполняет поиск асинхронно. Мысленно прокрутите программу: компьютер выполняет первую строчку, переходит ко второй, затем к третьей, и так далее до конца. Конечно, иногда программа перескакивает с места на место, но смысл в том, что в один момент времени исполняется одна строка кода.

У *Beagle* есть два способа поиска: синхронный и асинхронный. Первый означает «выполняй поиск, а я буду ждать его окончания», а второй – «начинай поиск, а я продолжу выполнение программы; когда закончишь, прерви меня». Они также известны как блокирующий и неблокирующий, потому что синхронный метод блокирует выполнение следующей строки, пока сам не завершит работу, в то время как асинхронный метод позволит вам выполнять код, а сам будет работать параллельно. »

Скорая помощь

Оператор += позволяет вам «подписывать» методы на события, а обратный ему -= отсоединяет методы.

» Вам необходимо это знать, чтобы учесть такую возможность: вдруг ваша программа завершится раньше, чем асинхронный метод успеет что-либо сделать? То есть завершится до того, как *Beagle* найдет какое-либо совпадение? Ответ состоит в том, что программа успешно отключится, и вызванный метод сгинет, не успев ничего напечатать. Вот почему мы не получили никакого результата, и, значит, нам надо ждать, пока *Beagle* завершит поиск.

Вот здесь на сцену и выходит *GTK*. Когда вы используете программы с графическим интерфейсом, приложение ждет, пока вы что-нибудь сделаете: выберете пункт меню или хотя бы наведете курсор мыши на кнопку. Фактически, любое действие посылает сигнал приложению, чтобы оно смогло отреагировать, то есть приложение просто ждет сигнала. Очевидно, такие приложения не отключаются, выполнив то или иное действие, иначе, например, в *Abiword* вам пришлось бы непрерывно что-то печатать, чтобы редактор не закрылся. Вместо этого они используют так называемый главный цикл [сообщений], который выглядит примерно так:

```
while (1) {
  LookForSignals();
  ActOnSignals();
}
```

Вы, конечно, сообразили, что это бесконечный цикл, но он практически не потребляет процессорное время и позволяет другим частям приложения (графическому интерфейсу или, в нашем случае, *Beagle*) работать на полную катушку. Нам того и надо, вот почему мы собираемся сесть на главный цикл *GTK*: пусть наша программа простаивает, пока *Beagle* производит поиск и вывод результата. Чтобы это сделать, добавим в конец метода Main() две строчки:

```
Gtk.Application.Init();
Gtk.Application.Run();
```

Этот код велит *GTK* запуститься и работать. Скомпилируйте программу, нажав F8, и запустите ее из командной строки. На этот раз вы должны получить список результатов, похожих на следующие:

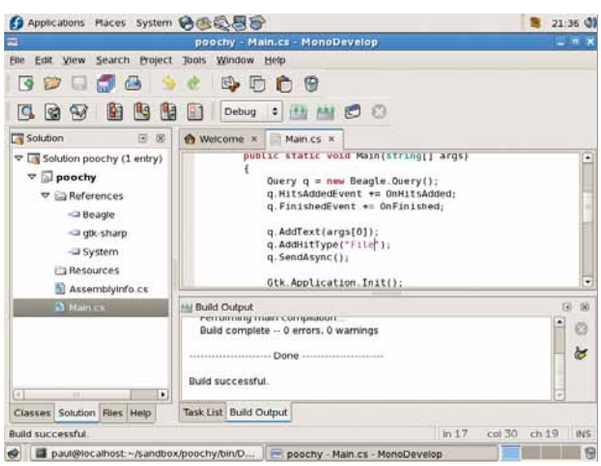
```
Hit: file:///home/paul/foo.xml
Hit: file:///usr/share/doc/tzdata-2006m/tz-link.html
Hit: file:///usr/share/gtk-doc/html/libuser/libuser-config.html
Hit: file:///usr/share/doc/nant-0.85/releasenotes.html
Hit: file:///usr/share/gtk-doc/html/libgnome/libgnome-gnomeconfig.
html
Hit: file:///usr/share/gtk-doc/html/gtk/gtk-question-index.html
Hit: file:///usr/share/gtk-doc/html/gtk/gtk-Resource-Files.html
```

Скорая помощь

После вызова Application.Quit() управление передается обратно методу Main(), где продолжается выполнение сразу после строки Application.Run(). Метод Quit() просто велит GTK завершить работу – вы вольны продолжить делать, что захотите.

Два шага вперед, один назад

Кроме основного результата, мы также получили побочный: выхода из приложения не происходит. Когда *Beagle* вернет все возможные результаты, *Poochy* будет по-прежнему чего-то ждать. Вы видели, что основной цикл приложения – бесконечный, и коль скоро мы вызвали метод Application.Run(), то вошли в состояние вечного исполнения. Пока программа не даст сбой, машина не перезагрузится или мы не нажмем Ctrl+C, *Poochy* так и будет находиться в этом состоянии.



» Можете писать свой код в *MonoDevelop*, но раз надо передать Poochy параметры, лучше запускать его из командной строки.

Часто задаваемые вопросы

Mono FAQ – часть I

Если тонкие моменты Mono вызывают у вас вопросы, то вот ответы на них...

» Не понял. Зачем мы пометили Reference, а потом еще и прописали строку using?

Добавление чего-то через Reference позволяет воспользоваться соответствующим кодом; добавление чего-то с помощью строки using позволяет сэкономить набор кода при вызове методов. Для подключения *Beagle* необходимо добавить его как Reference – тогда станут доступны его объекты и методы, и мы можем немедля их использовать, но только под полным именем, включающим ссылку на пространство имен. Строка using Beagle; сообщает .NET, что когда мы пишем Query, подразумевается Beagle.Query. В нашем случае экономия невелика; а вот, например, алгоритм SHA1 в .NET находится в пространстве System.Encryption.Cryptography, и пришлось бы писать System.Encryption.Cryptography.SHA1CryptoServiceProvider foo=new System.Encryption.Cryptography.SHA1CryptoServiceProvider() – согласитесь, довольно утомительно! Вместо этого можно поместить в начале using System.Encryption.Cryptography;, затем SHA1CryptoServiceProvider foo=new SHA1CryptoServiceProvider(). Помните, что *MonoDevelop* по умолчанию добавляет System в Reference.

» Почему мы используем +=, чтобы добавить метод к событию *Beagle*? Я думал, что += для добавления значения к переменной.

Да, += используется для сложения переменных, но почему бы не расширить эту метафору на события, происходящие во время работы *Beagle*, и не использовать += для добавления методов. Это также известно как подписка. Используя +=, вы можете 'подписать' несколько методов на одно событие – Mono просто запустит их по одному последовательно в порядке их добавления.

» Зачем нужен файл AssemblyInfo.cs?

Вы будете смеяться, но там и правда информация о вашей сборке! В терминах .NET сборка может быть как разделяемым объектом (SO файл в Linux или DLL в Windows), так и исполняемым файлом. Номер версии вашего исполняемого файла, имя программиста и другая информация находится в конечном двоичном файле **poochy**, и все это вы устанавливаете в **AssemblyInfo.cs**.

» Надо ли располагать открывающую и закрывающую фигурные скобки на отдельных строках?

Нет – вовсе нет! Лично я люблю ставить { на той же строке, что и оператор, ей предшествующий. Этот стиль известен как Единственно Верный Скобочный Стиль, потому что его приняли Брайан Керниган и Деннис Ритчи, когда они изобрели язык программирования C. Однако *MonoDevelop* по умолчанию использует стиль BSD, когда каждая скобка располагается на своей строке. Выбирайте, что вам больше нравится, главное – соблюдать единый стиль во всей программе!

Хитрые штучки

Если вы следовали первой части руководства этой серии, то у вас уже есть целый набор библиотек – Avahi, D-BUS, *GTK#*, ipod-sharp и другие. Теперь, узнав, как пользоваться *Beagle*, можете перенести полученные знания на другие библиотеки и удивить себя собственной элитной квалификацией! Но имейте в виду: документация очень скудна. *Beagle*, при всей нехватке его документации, еще много лучше прочих. Я рассмотрю некоторые библиотеки в следующих выпусках, а сейчас продолжайте читать...

Может, вам того и надо – к примеру, вы хотите периодически проверять ресурс или дожидаться, пока данные поиска доберутся до вас через сокет. Но *Poochy* задуман как программа, выполняющая поиск по требованию, а затем завершающаяся. Мы уже написали заглушку метода OnHitsAdded, а теперь займемся методом OnFinished(), который покамест пуст.

Когда *Beagle* вернет все результаты, он посмотрит, зарегистрирован ли метод для свойства FinishedEvent. Мы уже делали это с нашим методом OnFinished(), который вызывается по окончании работы *Beagle*. А сейчас нам нужно сообщить *GTK*, чтобы программа покинула основной цикл, тогда наше приложение сможет корректно завершиться. Это довольно просто – вот как выглядит новый метод OnFinished():

```
static void OnFinished(FinishedResponse response)
{
  Application.Quit();
}
```

Application.Quit() – это *GTK*-метод, который прерывает главный цикл, очищает все использованные *GTK*-ресурсы (в нашем случае их немного, так как мы не работаем с графикой), затем передает управление нашему приложению сразу же после строки Application.Run() в методе Main(). Поэтому путь работы программы будет следующим: Main() > SendAsync() > Application.Run() > основной цикл *GTK* > OnFinished() > Application.Quit() > Main().

Скомпилируйте, запустите и восхититесь собственным умом: ваша программа делает все, что задумано, и потребовала всего десять строк значащего кода!

Poochy выводит на экран все URL, RSS-ленты, документы *OpenOffice.org*, MP3, презентации PowerPoint, заметки Tomboy, исходный код, JPEG файлы, приложения... (глубокий вдох)... email, видео, встречи, zip-файлы и установленные пакеты на вашей системе, которые соответствуют запросу, передаваемому в AddText(). А если вас интересуют только файлы, находящиеся на вашем диске, а не все что ни попадя?

Погодите... еще не все!

Два урока назад мы рассматривали метод EndsWith(), применимый к строкам, который принимает строку в качестве аргумента и возвращает true, если строка A оканчивается строкой B. Например:

```
string foo = "bar";
bool baz = foo.EndsWith("r");
```

Когда этот код выполнится, переменная baz установится в **true**, так как foo оканчивается на 'R'. В .NET также есть метод StartsWith(), который действует наоборот – возвращает **true**, если одна строка начинается с другой. Выходит, мы можем заставить наш код выводить только файлы, поменяв метод OnHitsAdded() на следующий:

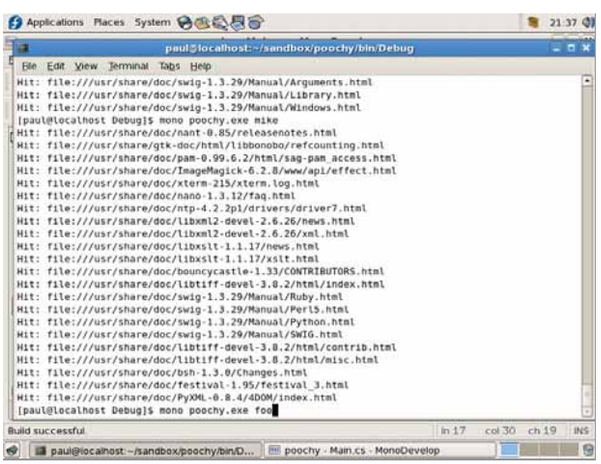
```
static void OnHitsAdded (Beagle.HitsAddedResponse response)
{
  foreach(Hit hit in response.Hits)
  {
    if (hit.Uri.StartsWith("file"))
    {
      Console.WriteLine("Hit: " + hit.Uri);
    }
  }
}
```

Это метод грубой силы: *Beagle* ищет все подряд, а мы фильтруем результаты перед выводом. Но *Beagle* можно научить искать и конкретный тип совпадений, с помощью метода AddHitType(): он позволит описать то, что вы ищете, обычным текстом. Корректные параметры включают Application, Calendar, Contact, Feeditem (для RSS), Image, IMLog, MailMessage и – вот оно! – File.

Вернувшись к методу Main(), найдите строку q.AddText(args[0]) и добавьте перед ней строку:

```
q.AddHitType("File");
```

Запустите программу, и увидите, что выводятся только файлы – никакой переписки из чатов, засоряющей результаты! LXF



› Готовый продукт: да, снова консольное приложение, но до чего ж крутое – чего только оно ни умеет!

Скорая помощь

Строки «using» позволяют писать меньше кода, но иногда способствуют коллизиям. Например, .NET содержит класс Timer в System.Timers, а SDL.NET содержит собственный класс Timer. Если вы объявите эти пространства с помощью выражений «using», то Mono не поймет, который Timer вы имели в виду, когда писали Timer foo=new Timer(); Придется указать это явно: написать, допустим, Timers.Timer.

Через месяц Мы построим пользовательский интерфейс с помощью Stetic!

Grub: Настройте

Один дистрибутив загрузить легко – но что если вам надо жонглировать дюжиной систем? **Ник Вейч** грубо прикидывает…

Наш эксперт

Ник Вейч
Ежедневно пользуясь Linux больше семи лет, Ник знает про эту систему все. Иногда он даже может припомнить, что именно.

Добро пожаловать на очередной урок по Linux. Сегодня я проведу несколько упражнений с загрузкой системы: покажу вам, как настроить её и улучшить с помощью *Grub*. Но сначала проведем разминку. Как вы уже знаете, при загрузке в вашу любимую операционную систему компьютер проходит несколько этапов. Все их понимать не обязательно, но в определенный момент времени прошитое ПО компьютера (BIOS) пытается загрузить ОС с диска (или другого устройства), используя для этого MBR (главную загрузочную запись) диска. Это небольшая область на диске, содержащая код, который найдет и загрузит ядро вашей ОС.

Долгое время Linux обходился системой *Lilo* (Linux Loader), устанавливавший код начальной загрузки в MBR и создающий map-файл с указаниями на код, который надо загрузить. У *Lilo* много ограничений, и не самое мелкое из них состоит в том, что этот файл нужно заново создавать всякий раз, когда вы хотите поменять инструкции загрузки.

Чтобы сделать загрузку более гибкой и внести новые возможности в этот процесс, был создан *Grub*, Grand Unified Boot manager (Большой Унифицированный менеджер Загрузки); он также поддерживает загрузку нескольких систем. Позор, что почти никто не использует его возможности! Среднестатистический дистрибутив Linux при установке создает загрузочную запись для себя, а если у вас есть Windows, то везучим он создаст запись и для него. Но может-то он гораздо больше! А тем, кто хочет загружать более одной системы, просто следует этим воспользоваться! Встрепенулись? Значит, можно начать урок.

Grub управляется файлом настройки, обычно находящимся в **/boot/Grub/menu.lst**. Когда *Grub* инициируется кодом начальной загрузки, он ищет этот файл, считывает его и выполняет указанные в нем команды. Преимущество такой системы в том, что после изменения конфигурации вам не нужно пересоздавать никакие файлы – с *Grub* вы даже можете менять процесс загрузки на лету!

На этом уроке!

- Коллекция экранов загрузки
- Многозагрузочные системы
- Защита паролем
- Аварийные загрузочные записи

Файл настройки

По умолчанию файл настройки *Grub* выглядит примерно так:

```
# grub.conf созданный anaconda
default=0
timeout=5
splashimage=(hd0,10)/boot/grub/splash.xpm.gz
hiddenmenu
title Fedora Core (2.6.18-1.2849.fc6)
        root (hd0,10)
        kernel /boot/vmlinuz-2.6.18-1.2849.fc6 ro
root=LABEL=/1 rhgb quiet
        initrd /boot/initrd-2.6.18-1.2849.fc6.img
title Fedora Core (2.6.18-1.2798.fc6)
        root (hd0,10)
        kernel /boot/vmlinuz-2.6.18-1.2798.fc6 ro
root=LABEL=/1 rhgb quiet
        initrd /boot/initrd-2.6.18-1.2798.fc6.img
title Wintest
        rootnoverify (hd0,5)
        chainloader +1
```

В начале файла идет набор глобальных настроек, описывающих загрузку по умолчанию, время до начала загрузки по умолчанию, экранную заставку, и опция, позволяющая спрятать меню, если не будет нажата какая-либо клавиша. Остальные настройки – индивидуальные записи меню, каждая из которых начинается с заголовка.

Начинаем

ОК, теперь взглянем на стандартные шаги, необходимые для загрузки с помощью *Grub*. Первым делом познакомьтесь с командой root. Она устанавливает корневое устройство, а также подключает его, чтобы добыть информацию о размере раздела и т.п. Вот как выглядит типичная запись:

```
root (hd0,0)
```

Команда устанавливает корневым разделом первый раздел первого диска. Если как корневой вы хотите использовать не-Linux раздел, то используйте команду **rootnoverify**, которая не пытается сначала подключить устройство. Вам необходимо это сделать, чтобы прицепить загрузчик Windows, например, который наверняка располагается на разделе NTFS.

Месяц назад Нейл Ботвик компилировал собственное ядро Linux. Класс!

LINUX FORMAT на DVD
» Grub 1.95

свой загрузчик

То, как *Grub* определяет диски, нередко вызывает путаницу (да такую, что в грядущем *Grub2* это решено переделать). Во-первых, он, не в стиле Linux, использует номера дисков вместо букв, так что согласно *Grub* **hdb** становится **hd1**. Вторая особенность состоит в том, что вся нумерация начинается с нуля!

Для вящего замешательства, если в Linux, пытаясь указать файл на втором разделе второго диска на системе, вы использовали **hdb2**, то в *Grub* это будет (**hd1,1**). Ветеран вы или салага, но когда-нибудь здесь непременно споткнетесь!

Определив корневое устройство, мы должны указать расположение образа, который хотим запустить. Это делается с помощью команды *kernel*. Она принимает имя файла (который представляет образ загружаемого ядра). *Grub* автоматически может распознавать тип файла для ядер Linux, а также большинства других Unix-систем.

Обычно вы видите примерно следующее:

```
kernel /boot/vmlinuz-2.6.18
```

Заметим, что здесь используется устройство, которое вы определили как корневое. Так что, увидев эту строчку, *Grub* попытается загрузить файл **(hd0,0)/boot/vmlinuz-2.6.18**.

Если образ ядра почему-либо находится на другом устройстве, то вам необходимо только приставить спереди нужный дескриптор устройства. Все остальные аргументы команды kernel передаются ядру, так что если ваши действия как-то связаны с конкретным драйвером, необходимые опции можно указать в конце этой строки.

> «Ветеран вы или салага, но когда-нибудь здесь да споткнетесь!»

Если ваше ядро использует образ RAM-диска для начальной загрузки, то вам необходимо указать его здесь, вот так, например:

```
initrd /boot/initrd-2.6.18
```

Вот и все, что нужно для стандартной записи в меню. Если бы вы набирали эти строки из командной строки *Grub*, то затем понадобилось набрать команду boot, однако ее наличие в меню подразумевается.

Чтобы запустить систему Windows, необходимо вызвать загрузчик Windows. Команда chainloader просто загружает другой загрузчик с указанного корневого раздела. Значение **+1** означает, что дополнительный загрузчик сидит в первом блоке раздела (именно туда его помещает Windows).

Двойной пакет Окон

Grub также позволяет установить несколько версий Windows. Стандартный загрузчик Windows 9x/DOS срабатывает, только если он находится на первом диске, что, очевидно, затрудняет установку более чем одной копии системы. Хуже того: если у вас на одном диске два загрузочных раздела Windows, могут возникнуть проблемы.

Пора уже сообразить, что *Grub* может помочь и в этой ситуации. Команда map позволяет переупорядочить разделы диска, и вы сможете установить две версии Windows, обманув загрузчик: он поверит, что грузится с первого диска. Вот как можно это сделать:

```
title Windows A
        hide (hd0,1)
        rootnoverify (hd0,0)
        chainloader+1
title Windows B
        hide (hd0,0)
        rootnoverify (hd0,1)
        chainloader+1
title Windows C
        map (hd0) (hd1)
        map (hd1) (hd0)
        rootnoverify (hd1,0)
        chainloader+1
```

Команда hide полностью прячет раздел от операционной системы и тем самым предотвращает возможные инциденты.

› Для вызова командной строки жмите 'c' в меню (если оно не заблокировано! См. указания по безопасности). Команды Grub будут доступны в интерактивном режиме.

Аварийное восстановление

Самое тяжкое преступление для рабочей системы – невозможность перезагрузиться после сбоя. Меньше всего вам хотелось бы, чтобы ядро системы впадало в панику из-за выхода из строя какого-то оборудования – особенно если вы администрируете удаленный сервер за тысячи километров от вас. *Grub* здесь может прийти вам на помощь: у него есть система восстановления после сбоев, позволяющая определить альтернативный порядок загрузки, если отказала основная последовательность инструкций.

Заметим, что отказ может быть как проблемой *Grub*, так и проблемой системы – например, с настройкой *Grub* все в порядке, но ядро не может загрузиться, не находит файловую систему или что там еще надо. Понятно, что защиты от дурака нигде не бывает, случаются такие обстоятельства, что процесс загрузки просто зависает, а не отказывает, однако данный метод позволит предотвратить множество типичных проблем. »

Скорая помощь

Если ваша сконструированная запись в меню не работает, можете ее отредактировать во время загрузки, для этого просто выберите нужную запись и нажмите 'e'. Выбор записи для редактирования производится клавишами курсора. Измените запись и нажмите Enter. Если все в порядке, нажмите 'b', и запись загрузится (если загрузка даст сбой, то вы вернетесь к меню).

Скорая помощь

Чтобы войти в командную строку, нажмите ‘c’ в меню *Grub*. Большинство команд *Grub* можно вводить прямо в ней. Чтобы вернуться в меню, просто нажмите Escape.

Вот пример простой настройки аварийной загрузки:

```
default saved
timeout 10
fallback 1 2

title Default system
root (hd0,0)
kernel /boot/vmlinuz-2.6.18
initrd /boot/initrd-2.6.18
savedefault fallback

title Failsafe mode
root (hd0,0)
kernel /boot/vmlinuz-2.6.18 failsafe
initrd /boot/initrd-2.6.18
savedefault fallback

title Alternative
root (hd0,3)
kernel /boot/vmlinuz-2.4.34
initrd /boot/initrd-2.6.18
savedefault
```

Строка **default** велит *Grub* искать сохраненное значение. Строка **fallback** определяет порядок альтернативной загрузки (умолчание по умолчанию 0), поэтому в данном примере сначала пробуется запись Failsafe, затем запись Alternative.

Grub ищет в своем каталоге файл **default**, содержащий значение загрузки записи по умолчанию. Это значение можно изменить командой *Grub-set-default*, входящей в состав *Grub*, но так как многие дистрибутивы ее не включают, то можете сделать, например, так:

```
echo ‘1’ > /boot/grub/default
chmod 600 /boot/grub/default
```

Учтите, что после того, как диск произвел аварийную загрузку, значение по умолчанию изменится на первую успешную загрузочную запись, поэтому вам необходимо обновить значение в файле ‘default’ и вернуть все на свои места.

```
  Booting 'default'

root (hd1,4)

Error 21: Selected disk does not exist
  Booting 'Fedora Core (2.6.18-1.2798.fc6)'

root (hd0,0)
 Filesystem type is ext2fs, partition type 0x83
kernel /vmlinuz-2.6.18-1.2798.fc6 ro root=/dev/VolGroup00/LogVol
   [Linux-bzImage, setup=0x1e00, size=0x1b94fc]
initrd /initrd-2.6.18-1.2798.fc6.img
   [Linux-initrd @ 0x1fcce000, 0x211edb bytes]
savedefault

Uncompressing Linux... Ok, booting the kernel.
ACPI: Getting cpuindex for acpiid 0x1
ACPI: Getting cpuindex for acpiid 0x2
ACPI: Getting cpuindex for acpiid 0x3
powernow-k8: BIOS error - no PSB or ACPI _PSS objects
Red Hat nash version 5.1.19 starting
  Reading all physical volumes.  This may take a while...
  Found volume group "VolGroup00" using metadata type lvm2
  2 logical volume(s) in volume group "VolGroup00" now active
```

› Система аварийной загрузки в действии: первый образ почему-то не загрузился, и *Grub* немедленно переходит к следующей записи в списке. Сам!

Ставим замок

Навидавшись чудес, совершаемых с помощью *Grub*, вы можете обеспокоиться, нельзя ли все это использовать в нехороших целях. Что ж, ваш компьютер никогда не находится в полной безопасности, если он, конечно, не защищен физически, однако существуют меры по усилению сохранности системы.

Grub поддерживает защиту паролем как в виде обычного текста, так и в виде md5-шифрованных паролей.

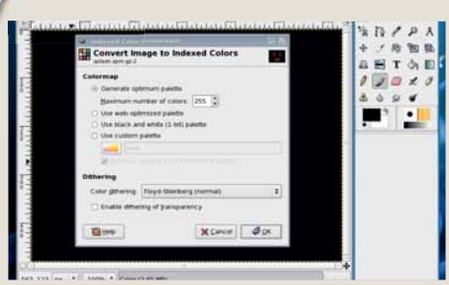
Я покажу, как это работает. Создайте зашифрованный пароль прямо в *Grub* как в терминале (от имени root).

```
grub
grub> md5crypt
Password: ********
Encrypted:$1$lsFjn1$VROhkBS.BaraJ6brXjnc8.
```

Теперь добавьте следующий код в первый раздел вашего файла настройки **menu.lst**:

```
 # заголовок
# обычно создается
# вашим дистрибутивом
default =0
timeout=5
splashimage=(hd0,0)/boot/grub/splash.xpm.gz
password -md5 :$1$lsFjn1$VROhkBS.BaraJ6brXjnc8.
hiddenmenu
title Linux-one
        root(hd0,0)
```

Шаг за шагом: Настраиваем заставку

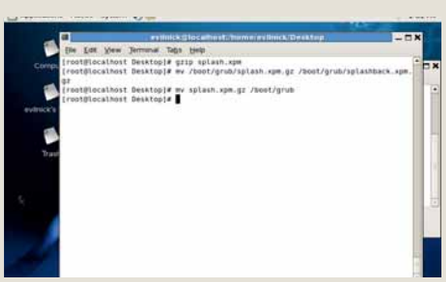


» Новый пейзаж на экране

Вам понадобится изображение размером 640*480. Сохраните его как **splash.xpm** в формате XPM. Теперь откройте терминал и сожмите файл с помощью *gzip*:

```
gzip splashscreen.xpm
```

Создастся файл, оканчивающийся на ‘.gz’.



» Установите картинку

Сохраните копию **splash.xpm.gz** из **/boot/grub** (потребуются права root), а затем установите новую заставку с помощью

```
mv /boot/grub/splash.xpm.gz /boot/grub/splashbak.xpm.gz
mv splash.xpm.gz /boot/grub
```

» Перезагрузитесь

Перезагрузитесь и наслаждайтесь новой заставкой. Если у вас еще нет файла **splash.xpm.gz**, то, возможно, ваш дистрибутив не включает версию *Grub*, использующую заставки – тогда соберите его сами из исходных текстов!

```
...
```

и т.д. Далее, существует несколько способов использования паролей. В показанном выше файле, пользователь сможет выбирать пункты меню, но не запускать команды *Grub* через командную строку и не редактировать существующие настройки. Если попытаетесь, то вас попросят пароль – тех, кто его не знает, *Grub* пнет обратно в меню.

Это довольно простая мера защиты; а если вы хотите вообще скрыть присутствие альтернативных записей загрузки? Что ж, команда password может принимать еще один аргумент, указывающий на другое меню *Grub*.

```
password -md5 :$1$lsF… /grub/menu2.lst
```

Вы можете, конечно, назвать файл как угодно и даже хранить его где угодно, лишь бы *Grub* мог до него добраться.

При загрузке *Grub* отобразит обычное меню, с теми же ограничениями. А вот если на экране выбора в меню вы нажмете ‘p’ и введете правильный пароль, то он загрузит совершенно новый файл меню со всеми спрятанными опциями.

И наконец, полезный способ использовать пароль – это применить его к отдельному образу. Строка с паролем вставляется сразу после заголовка title. Когда данная запись выбирается в меню, *Grub* потребует пароль – а не то откажется продолжить работу. Это полезно, если к вашей системе имеет доступ несколько групп пользователей.

Создание загрузочного Grub CD

Grub на загрузочном диске может очень пригодиться – например, чтобы оживить систему, которая перестала работать, или для загрузки с устройства, не распознаваемого BIOS (это непросто, так как придется установить ядро, поддерживающее устройство в образе RAM-диска, и использовать его для загрузки настоящего ядра).

Кроме загрузки с неподдерживаемого устройства, вы также можете использовать спасательный диск, чтобы переустановить первый этап *Grub* в главной загрузочной записи на диске.

Чтобы загрузиться с CD, *Grub* пользуется образом El Torito. El Torito – это имя стандартного протокола загрузки с CD (*странное? См. врезку Знаете ли вы?.. ниже*), которому требуется особый код.

Создайте каталог, где будет хранится образ, и реплицируйте структуру каталога boot следующим образом:

```
mkdir -p grubcd/boot/grub
```

Теперь необходимо скопировать нужный образ *Grub*. Ваш дистрибутив обычно хранит их в **/usr/share/Grub/i386-pc/**, хотя имя последнего каталога может варьироваться. Для Fedora, например, мы использовали

```
cp /usr/share/grub/i386-redhat/stage2_eltorito grubcd/boot/grub/
```

Без конфигурационного файла, *Grub* просто загрузит командную строку, позволив вам вводить команды загрузки напрямую. Однако вы можете при желании включить такой файл (например, списать его с вашей основной системы Linux, чтобы использовать его как резервную копию). И хотя вы создаете *Grub* CD, это не значит, что на него нельзя поместить другие полезные вещи.

› Проблемы с загрузкой? Указали не тот диск в настройке? Говорили вам: споткнетесь! Нажмите ‘e’ и исправьте загрузочную запись во время сессии.

Для создания образа можно использовать команду *makeiso*. Нужно будет сообщить ей, где находится загрузочный образ. Затем, так как данный образ *Grub* не использует систему эмуляции диска (где образ диска эмулирует загрузочную дискету), на это также придется указать, да еще добавить несколько дополнительных параметров для максимальной совместимости с системами со старым BIOS. Сведя все вместе, получаем, что ввести надо

```
mkisofs r -b grubcd/grub/stage2eltorito -no-emul-boot -boot-loadsize
  4 -boot-info-table -o Grubcd.iso
```

Данная команда создаст файл ISO, который будет грузиться (проверьте, что BIOS настроен на загрузку с CD!). Осталось использовать *CDRecord* или инструмент вроде *K3b* для прожига образа на диск, и все готово.

Завершение

Наш урок окончен, и настало время остыть. Если ваши Linux мускулы еще не болят, то можете попытаться двигаться дальше. *Grub* хорош, но разве вы не думаете, что он может больше? Разве вас не достает, как он обзывает диски и разделы? Многие уже на это натыкались. На самом деле их так много, что *Grub2*, который уже довольно долгое время находится в разработке, исправит эти недочеты.

Grub2 вырос из старого исследовательского проекта Pupa (Preliminary Universal Programming Architecture), который был создан для исследования процесса обновления. Некоторые новые возможности *Grub2* включают расширяемую, модульную среду, поддержку скриптов и ориентированность на многие платформы.

Хотя система следующего поколения еще в разработке, пользоваться ею уже можно. Вы можете ознакомиться с ходом разработки *Grub2* в wiki по адресу **http://grub.enbug.org**. LXF

Знаете ли вы?..

Имя El Torito (это раширение формата ISO9660, позволяющее загружать компьютер с DVD и CD дисков) по-испански значит «бычок». Механизма загрузки оно никак не проясняет. Но так называется сеть ресторанов мексиканской кухни в США. Легенда гласит, что стандарт загрузки обдумывался как раз в одном из этих ресторанов в Калифорнии.

Hajime Nakano

› Спецификацию загрузки спроектировали, поев похожей пищи.

» Через месяц Мы вгрыземся в создание собственных .deb пакетов!

Glade 2.x — визуальная

ЧАСТЬ 4 Красивый интерфейс можно запрограммировать, вдалбливая бесконечные строчки кода, а можно – легким движением руки, в которой зажата мышь. **Андрей Боровский** рассматривает второй способ.

Каждый уважающий себя набор визуальных элементов располагает средством для того, чтобы бросить пару-тройку компонентов на заготовку окна приложения, настроить их свойства и получить остов будущей программы. У *Qt* есть *Qt Designer*, у *wxWidgets* есть *DialogBlocks*, а у *GTK+* есть *Glade*. Если ваш опыт программирования ограничивается такими средами, как Borland C++ Builder или Microsoft Visual C#, приступая к программированию с помощью *Glade*, вы должны понять одну важную вещь: *Glade* – это не IDE. В нем нет ни интегрированного компилятора, ни интерактивного отладчика, ни редактора кода с автоматическим завершением строки. *Glade* 2.x генерирует минимальный код, который вам придется дописывать в вашем любимом редакторе, а *Glade* 3.x вообще стремится возложить генерацию кода на другие инструменты. Несмотря на то, что в ближайшее время старый *Glade* будет вытеснен третьей версией, знакомство с этим инструментом разработки позволит нам взглянуть на структуру сложных программ *GTK+* глазами самих авторов набора компонентов. Поэтому данная статья посвящена *Glade 2.x*, а в следующей статье мы узнаем, чем *Glade 3.x* отличается от своего предшественника. Мы воспользуемся *Glade* для создания простой программы ImageViewer. Ее исходные тексты можно найти на диске, но я советую вам пройти вместе со мной все шаги по созданию данного приложения.

Каждый раз, когда мы создаем новый проект *Glade*, перед нами появляется диалоговое окно, в котором нам предлагается выбрать тип проекта – *GTK+* или GNOME (**Рис. 1**). Нам нужен *GTK+*.

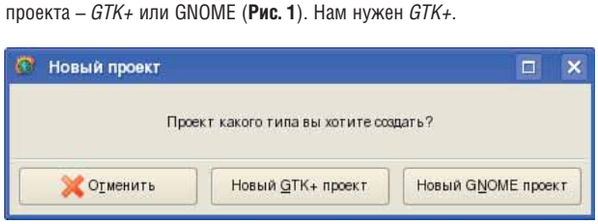


Рис. 1 Окно выбора типа проекта.

В среде *Glade* в процессе работы может быть открыто несколько окон, постоянными из которых являются три: главное окно, окно палитры и окно редактора свойств объектов. Главное окно (**Рис. 2**) содержит меню и панель инструментов для управления проектом. В рабочей части окна расположен список визуальных элементов верхнего уровня.

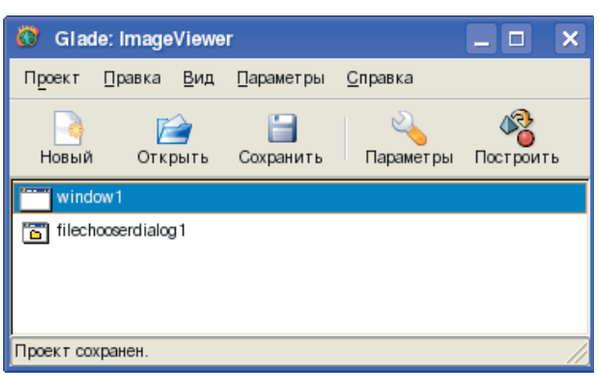


Рис. 2 Главное окно Glade.

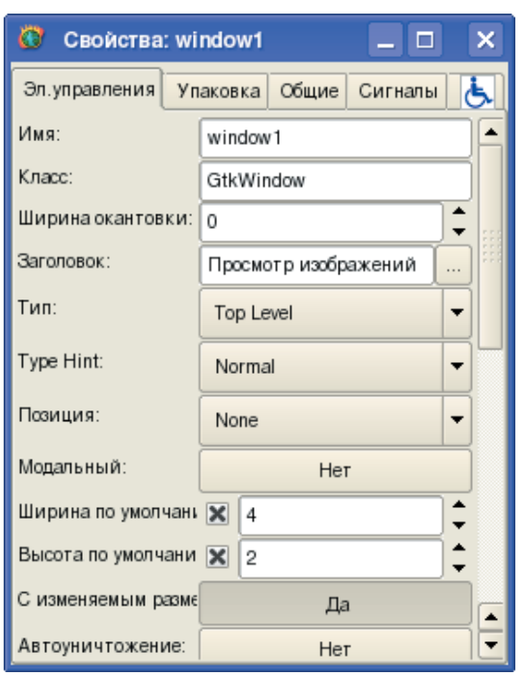
Окно «Палитра», которое, в том или ином виде, присутствует в каждой среде визуальной разработки, предоставляет нам доступ к пиктограммам визуальных элементов. Окно Палитра *Glade 2.** (**Рис. 3**) содержит несколько вкладок. Список вкладок зависит от типа проекта (*GTK+* или GNOME). При работе с нашим проектом палитра будет содержать три вкладки: GTK+ Основные, GTK+ Дополнительные и Устаревшие.

Окно редактора свойств (**Рис. 4**) предназначено, естественно, для редактирования свойств визуальных элементов. Это окно также содержит несколько вкладок. Вкладка Эл. управления позволяет настраивать специфические свойства визуального элемента. Вкладка Упаковка позволяет настраивать параметры упаковки тех визуальных элементов, которые вложены в какой-либо контейнер. С помощью вкладки Общие можно настраивать общие свойства визуальных элементов, а вкладка Сигналы предназначена для связывания сигналов и их обработчиков. Наконец, безымянная вкладка, пиктограмма которой изображает инвалидное кресло, позволяет настраивать параметры доступности

Месяц назад Мы разбирались с сигналами и событиями GTK+.

КЛАССИКА

(accessibility) визуальных элементов GNOME. Эту вкладку мы пока рассматривать не будем.



› Рис. 4 Окно редактора свойств.

Щелкните по элементу Окно на вкладке GTK+ Основные палитры компонентов. В результате в проект будет добавлен элемент верхнего уровня Окно (вы можете видеть его в списке элементов верхнего уровня в главном окне проекта), а перед нами откроется пустое окно со стандартным заголовком. Это окно станет главным окном создаваемого нами приложения. Программа *GTK+* может содержать несколько элементов верхнего уровня, и для каждого из них можно открыть независимое окно.

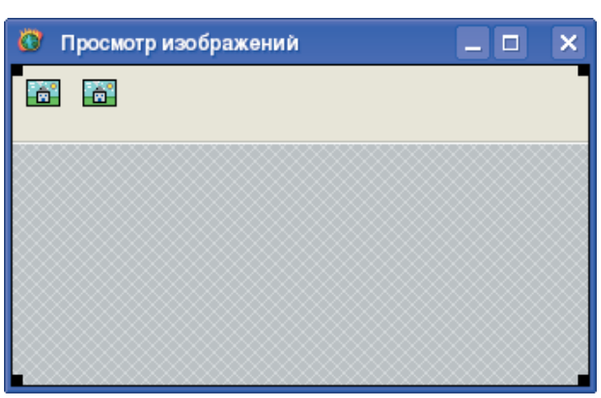
Давайте настроим параметры главного окна программы. Для того, чтобы открыть объект интерфейса верхнего уровня в редакторе свойств, нужно выделить этот объект в главном окне *Glade*. Перейдя в редакторе свойств на вкладку Эл. управления, мы можем указать текст заголовка окна (свойство Заголовок). Установите также флажки Ширина по умолчанию и Высота по умолчанию. В соответствующих наборных счетчиках укажите значения **200** и **150**. Таким образом вы зададите начальные размеры окна.

Вы, конечно, помните, что любое приложение *GTK+* с более-менее сложным интерфейсом должно использовать контейнеры для управления визуальными элементами. Разработка интерфейса в *Glade* также начинается с создания контейнера. Выберите на вкладке палитры GTK+ Основные элемент Вертикальный бокс. Этому элементу соответствует вертикальный контейнер GtkVBox. Перенесите вертикальный контейнер мышью в окно приложения. Во многих редакторах форм этот процесс называется перетаскиванием, поскольку компонент перетаскивается мышью с палитры компонентов в окно формы, но в *Glade* все происходит по-другому. Для того чтобы перенести компонент в форму, нужно сначала щелкнуть мышью по этому компоненту в палитре, а затем – по той области на форме, в которой этот компонент должен быть размещен (вот почему мы используем термин «перенос» а не «перетаскивание»). При переносе в форму вертикального контейнера открывается диалоговое окно, в котором *Glade* просит нас указать число строк в нем. Создайте контейнер с двумя строками. В результате главное окно будет разделено на две горизонтальные области, расположенные одна под другой (для вертикального расположения дочерних элементов). В верхнюю область перенесите с палитры объект Панель инструментов. Панель инструментов также представляет собой контейнер, так что при переносе панели в окно приложения мы снова должны указать количество дочерних элементов (на этот раз – кнопок панели). Панель инструментов нашего приложения должна содержать две кнопки. Вы можете заметить, что после переноса панели инструментов в окно формы, размеры верхней области вертикального контейнера подстраиваются под размеры панели, а на самой панели появляются две незаполненные области для размещения кнопок (**Рис. 5**). В эти области мы должны перенести два элемента Toolbar Button – кнопки панели инструментов, которым соответствует объект GtkToolButton. В окне панели инструментов появляются заготовки кнопок.

› (Рис. 3) Главное окно палитры.

Что в имени тебе моем

Поскольку на этом этапе мы уже работаем с несколькими объектами, нам следует познакомиться с понятием имени объекта. Каждому объекту *GTK+* может быть присвоено имя. Это имя хранится в одном из полей объекта, и его не следует путать с именем переменной, которая содержит указатель на объект (имя объекта, хранящееся в самом объекте, по праву можно назвать «именем собственным»). До тех пор, пока наши программы состояли из одного файла и все переменные, содержащие объекты, находились в одной области видимости, имена объектов не имели для нас значения. Однако в »



› Рис. 5 Главное окно *Glade*.

сложных проектах, состоящих из нескольких файлов, имена объектов *GTK+* играют важную роль, поскольку именно они служат для идентификации объектов. В редакторе свойств объектов на вкладке Эл. управления присутствует свойство Имя, которое позволяет вам получить доступ к имени объекта на этапе визуального программирования. Две кнопки, которые вы добавили на панель инструментов, имеют имена toolbutton1 и toolbutton2. Я предполагаю, что вы добавляли кнопки в порядке слева направо, то есть первой слева является кнопка toolbutton1.

Только что созданные кнопки выглядят совсем не так, как нам хотелось бы. Для того, чтобы придать кнопкам требуемый вид, мы должны отредактировать их свойства. Выберите в окне формы кнопку toolbutton1 и перейдите в редактор свойств, на вкладку Эл. управления. Свойству Метка присвойте значение Открыть. Затем щелкните по значку раскрывающегося списка в поле свойства Иконка, в открывшемся списке выберите пиктограмму Открыть. Нам необходимо создать заготовку обработчика для сигнала кнопки clicked. Перейдите на вкладку Сигналы редактора свойств и щелкните кнопку с многоточием справа от строки ввода Сигнал. Перед вами раскрывается список сигналов, в котором следует выбрать сигнал clicked. В строке Обработчик появляется имя обработчика сигнала (по умолчанию – on_toolbutton1_clicked). Щелкните кнопку Добавить. В результате новая функция будет добавлена в список обработчиков сигналов кнопки toolbutton1.

Перейдем теперь к редактированию свойств кнопки toolbutton2. Свойству Метка второй кнопки присвоим значение О программе. В списке значений свойства Иконка выберем значение Справка, затем создадим заготовку обработчика сигнала clicked (назовем его on_toolbutton2_clicked), так же как и в предыдущем случае.

В нижнюю область вертикального контейнера следует перенести элемент палитры Изображение (объект GtkImage, имя объекта по умолчанию – image1). Выделите объект image1 в окне формы. Перейдите на вкладку Упаковка редактора свойств и присвойте значение Да свойству Заполнение. Далее, выделите в окне формы палитру инструментов (объект toolbar1 типа GtkToolbar). Перейдите на вкладку Упаковка редактора свойств и также присвойте значение Да свойству Заполнение (**Рис. 6**).

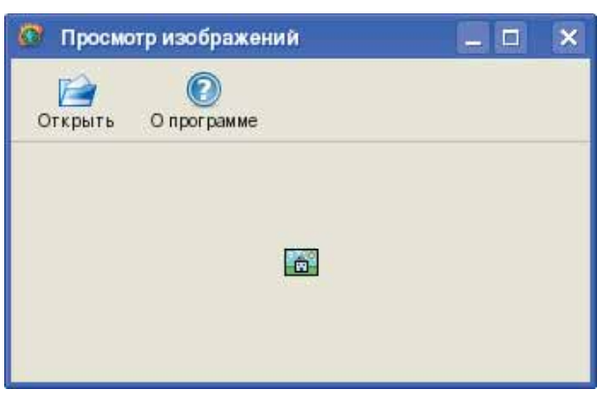


Рис. 6 Форма приложения с кнопками и объектом *GtkImage*.

Заголовок

Мы должны еще добавить в проект окно с информацией о программе. Это просто, поскольку необходимое окно входит в набор визуальных компонентов *GTK+* (**Рис. 7**). Вы найдете его на вкладке GTK+ Дополнительные палитры компонентов (элемент About Dialog, объект GtkAboutDialog). Окно с информацией о программе является элементом верхнего уровня, поэтому после щелчка мышью в палитре компонентов объект aboutdialog1 появится в списке элементов верхнего уровня. Перейдя на вкладку Эл. управления редактора свойств, вы сможете настроить основные свойства этого диалогового окна (название программы, имя автора, информацию о правах и т.д.). Никаких обработчиков сигналов для этого окна создавать не нужно, поскольку все необходимые обработчики создаются по умолчанию.

Рис. 7 Окно с информацией о программе.

Сохраним наш проект. Как уже отмечалось, *Glade* 2.* позволяет генерировать исходный код приложения. Для этого нужно щелкнуть на кнопку Построить. Теперь мы можем скомпилировать нашу программу (точнее, ту ее часть, которую мы создали). Перейдем в директорию проекта и выполним файл **autogen.sh**. В результате будут созданы сценарий configure и make-файл. Мы можем собрать программу с помощью команды **make**. Двоичный файл программы будет создан в директории **src**. Если вы запустите программу на этом этапе (я рекомендую первый раз запустить программу с консоли), то заметите две неприятные вещи. Во-первых, окно с информацией о программе само собой появляется вместе с главным окном программы. Во вторых, после закрытия главного окна программы ее работа не завершается (это заметно, если запустить программу из окна консоли). Для того, чтобы исправить ситуацию с окном aboutdialog1, нам нужно отредактировать файл **src/main.c**. Откройте этот файл в текстовом редакторе и удалите из него строки

```
aboutdialog1 = create_aboutdialog1 ();
gtk_widget_show (aboutdialog1);
```

Как вы, конечно, поняли, файл **main.c** содержит главную функцию программы. По умолчанию этот файл содержит код для создания и отображения на экране всех визуальных элементов верхнего уровня. Помимо файла **main.c** среда *Glade* создала еще несколько файлов исходных текстов в директории **src**. Файлы **interface.c** и **interface.h** содержат код, отвечающий за работу элементов управления, которые мы создали в режиме визуального программирования. Файлы **support.c/support.h** включают некоторые вспомогательные функции. Обработчики сигналов, которые мы создали в редакторе свойств *Glade*, объявлены в файле **callbacks.h**, а определены – в файле **callbacks.c** (вы можете изменить имена файлов исходных текстов, заданные по умолчанию, в окне настроек *Glade*). Только файлы **main.c** и **callbacks.c** предназначены для ручного редактирования. Текст остальных файлов перезаписывается *Glade* в процессе генерации кода. Теперь вспомним первые уроки программирования *GTK+*. Для того, чтобы программа завершалась вместе с закрытием главного окна, необходимо определить обработчик сигнала destroy объекта окна. В главном окне *Glade* выделите главное окно приложения и перейдите на вкладку «Сигналы» редактора свойств. Выберите в списке сигналов сигнал destroy (его не следует путать с доступным там же событием “destroy event”) и создайте заготовку обработчика для этого сигнала точно так же, как вы создавали заготовки обработчиков сигналов clicked. Сгенерируйте заново исходный код, нажав кнопку «Построить». Откройте в текстовом редакторе файл **callbacks.c**. Найдите заготовку обработчика on_window1_destroy() и добавьте в него вызов функции gtk_main_quit():

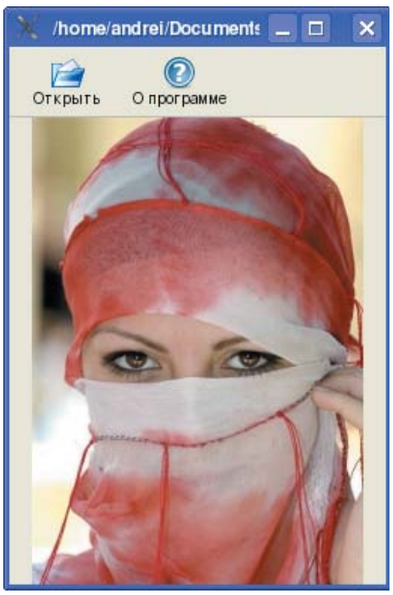
```
void on_window1_destroy (GtkObject * object, gpointer user_data)
{
  gtk_main_quit();
}
```

Мы используем тот же метод завершения программы, что и во всех предыдущих примерах. Сохраните файл **callbacks.c**. Теперь можете перекомпилировать и запустить программу снова. Окно описания программы уже не появляется, а закрытие главного окна программы приводит к ее завершению. После внесения изменений в свойства визуальных объектов вы должны каждый раз заново генерировать исходный код (и, конечно, сохранять все изменения, сделанные в исходных текстах вручную). Что произойдет, если вы сначала отредактируете код одного обработчика сигнала, а затем добавите новый обработчик в режиме визуального редактирования? Ничего страшного не случится. *Glade* добавит новый код в файл **callbacks.c**, и не тронет изменения, сделанные вами (этим свойством обладают только файлы **callbacks.c** и **main.c**).

Завершающая часть работы над нашей программой связана с добавлением кода в обработчики событий clicked двух кнопок. Сами функции-обработчики on_toolbutton1_clicked() и on_toolbutton2_clicked() уже определены в файле **callback.c**. Первая функция должна вызывать диалоговое окно выбора файла и загружать выбранный пользователем файл изображения в компонент image1:

```
void
on_toolbutton1_clicked (GtkToolButton * toolbutton, gpointer user_data)
{
  GtkWidget * file_chooser_dialog;
  file_chooser_dialog = gtk_file_chooser_dialog_new("Открыть", NULL,
    GTK_FILE_CHOOSER_ACTION_OPEN,
    GTK_STOCK_CANCEL, GTK_RESPONSE_CANCEL,
    GTK_STOCK_OPEN, GTK_RESPONSE_ACCEPT, NULL);
    if (gtk_dialog_run(GTK_DIALOG (file_chooser_dialog)) == GTK_
RESPONSE_ACCEPT)
    {
      GtkWidget * toplevel;
      GtkWidget * image;
      char *filename;
      filename = gtk_file_chooser_get_filename(GTK_FILE_
CHOOSER(file_chooser_dialog));
      toplevel = gtk_widget_get_toplevel(GTK_WIDGET(toolbutton));
      image = lookup_widget(GTK_WIDGET(toplevel), "image1");
      gtk_image_set_from_file(GTK_IMAGE(image), filename);
      gtk_window_set_title(GTK_WINDOW(toplevel), filename);
      g_free (filename);
    }
    gtk_widget_destroy (file_chooser_dialog);
}
```

Все необходимые заголовочные файлы включены в файл **callbacks.c**, так что мы можем о них не беспокоиться. Диалоговое окно выбора файла реализуется объектом GtkFileChooserDialog. Объект создается функцией gtk_file_chooser_dialog_new(). В первом параметре этой функции передается заголовок создаваемого окна. Второй параметр может содержать ссылку на родительский визуальный элемент. Третий параметр служит для передачи константы, которая указывает, для чего предназначено диалоговое окно – для открытия или для сохранения файла. Далее следует список параметров, состоящий из имен кнопок и констант, которые должны идентифицировать эти кнопки. Список может содержать описание одной или нескольких кнопок и должен заканчиваться значением **NULL**. Функция gtk_dialog_run() делает диалоговое окно видимым и возвращает управление после того, как пользователь нажмет одну из кнопок. При этом функция возвращает значение, сопоставленное нажатой кнопке при вызове функции gtk_file_chooser_dialog_new(). Если пользователь нажал кнопку «Открыть», имя которой задано в константе GTK_STOCK_OPEN, функция gtk_dialog_run() возвращает значение GTK_RESPONSE_ACCEPT. В случае выбора пользователем кнопки «Открыть» мы должны узнать имя выбранного файла, получить указатель на объект image1 и загрузить файл в этот объект. Если вас удивляет, что я включил получение указателя на объект image1 в список наших задач, то вспомните, что у вас нет доступа к переменной, которая хранит этот указатель. Для того, чтобы получить указатель на объект image1, мы должны воспользоваться функцией поиска объектов в иерархии визуальных компонентов нашего приложения. Мы начинаем поиск объектов с того, что находим визуальный элемент верхнего уровня. Указатель на визуальный элемент верхнего уровня мы получаем с помощью функции gtk_widget_get_toplevel(). В поисках нужного объекта функция gtk_widget_get_toplevel() перебирает иерархию визуальных элементов, начиная с того элемента, чей указатель передан ей в единственном параметре. В качестве стартового значения для поиска мы передаем функции gtk_widget_get_toplevel() указатель на объект-кнопку, для которой вызван обработчик on_toolbutton1_clicked() (адрес этого объекта передается функции обработчику в параметре toolbutton, и это единственный адрес объекта нашего приложения, которым мы располагаем в данный момент). Адрес объекта, представляющего элемент верхнего уровня (в нашем случае это объект window1 – главное окно приложения) мы сохраняем в переменной toplevel. Далее мы получаем указатель на объект image1 с помощью функции lookup_widget() (эта функция сгенерирована *Glade* для нашего проекта и объявлена в файле **support.h**). Функция lookup_widget() позволяет найти визуальный объект по его имени. Документация гласит, что первым аргументом lookup_widget() должен быть указатель на визуальный элемент верхнего уровня, либо указатель на любой другой элемент иерархии приложения. В качестве отправной точки поиска мы выбираем элемент верхнего уровня, во-первых, потому, что так красивее, и во-вторых, потому, что у нас все равно уже есть указатель на объект главного окна. Вторым аргументом lookup_widget() должно быть имя искомого визуального элемента. Указатель на объект image1 сохраняется нами в переменной image. Мы получаем имя выбранного пользователем файла с помощью функции gtk_file_chooser_get_filename() и загружаем хранящееся в файле изображение в объект image1 с помощью функции gtk_image_set_from_file(). Остальной код обработчика должен быть вам понятен. Осталось заполнить обработчик сигнала clicked кнопки toolbutton2:



› Рис. 8 Работающая программа *ImageView*.

```
void
on_toolbutton2_clicked(GtkToolButton   *toolbutton, gpointer user_data)
{
  GtkWidget * about_dialog;
  about_dialog = create_aboutdialog1();
  gtk_widget_show(about_dialog);
}
```

Получить указатель на объект aboutdialog1 очень просто. В файле **interface.h** определена функция create_aboutdialog1(), которая создает объект окна описания программы и возвращает указатель на него. Нам остается только вывести окно на экран с помощью gtk_widget_show(). Это те самые два вызова функций, которые нам пришлось удалить из файла **main.c**, для того, чтобы окно описания программы не появлялось тогда, когда не надо. Теперь мы можем проверить, как работает наша программа (**Рис. 8**). LXF

Через месяц Мы узнаем, что нового привносит Glade 3.x в процесс визуальной разработки.

Консольный

ЧАСТЬ 10 Привыкли считать консольные приложения лишенными пользовательского интерфейса? Напрасно – старый добрый терминал способен на многое. На что именно – расскажет **Андрей Боровский**.

Real computer scientists despise the idea of actual hardware.
Hardware has limitations, software doesn't.
It's a real shame that Turing machines are so poor at I/O.

Из программистского фольклора.

Каждому, кто приступает к изучению программирования для Unix, консольный ввод/вывод представляется простым делом. Для создания графических программ необходимо сначала изучить язык программирования, а затем – один из интерфейсов программирования графики. Для создания консольных программ достаточно изучить язык программирования. Если вы не шли к изучению программирования окольными путями, то ваши первые программы наверняка были консольными. Однако интерфейс консоли Unix обладает гораздо большим числом возможностей, чем кажется на первый взгляд. О некоторых дополнительных возможностях работы с консолью мы и поговорим сегодня. Богатство функций работы с консолью в Unix объясняется тем, что у Unix долгая история. Когда-то терминалы были настоящими устройствами (в некоторых областях человеческой деятельности такие устройства применяются и сейчас), подключенными к компьютеру через последовательный порт. Часто терминал и компьютер были отделены друг от друга прослойкой в виде пары модемов и телефонной линии. Последовательный порт и модем считались неотъемлемой частью терминала, и интерфейс управления ими стал частью интерфейса управления терминалом. Кроме того, такие устройства, как, например, принтеры, тоже считаются терминалами Unix. Если в графическом программировании принтер считается устройством ввода/вывода данных, подобным графическому дисплею, то вполне логично, что в консольном программировании принтер рассматривается как терминал. Таким образом, с точки зрения Unix терминалами считается множество устройств, которые работают совершенно по-разному. Неудивительно, что для управления всем этим многообразием устройств потребовался сложный интерфейс. И хотя вы вряд ли подключаетесь к своему компьютеру с помощью модема (у вас, возможно и модема-то уже нет), а для вывода документа на печать вы, скорее всего, используете фильтр PostScript, некоторые возможности управления терминалом, появившиеся в незапамятные времена, могут пригодиться и вашим программам.

Предотвращение перенаправления вывода

Вы, конечно, знаете, что в Unix вывод консольной программы, предназначенный монитору, может быть перенаправлен в файл или в поток ввода другой программы. Часто это бывает не только полезно, но даже необходимо. Теперь можете проверить свою сообразительность. Отодвиньте эту статью на десять секунд и подумайте, в каких случаях программа не должна допускать перенаправление своего вывода? Самый простой пример – интерактивная консольная программа. Интерактивные программы выводят данные небольшими порциями, после чего останавливаются в ожидании реакции пользователя. Примером такой программы может служить клиент SSH. Если пользователь пытается перенаправить вывод интерактивной программы, значит, он, скорее всего, просто не понял, как работать с вашей программой. Желательно предотвратить бессмысленное действие и сообщить пользователю об этом. Рассмотрим сначала простейший пример программы, которая не позволяет перенаправить свой вывод на устройство, не являющееся терминалом (текст этой программы вы найдете на диске, в файле **noredirect.c**).

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>

int main (int argc, char ** argv)
{
  char * errstr = "I will not redirect my output!\n";
  if (!isatty(fileno(stdout))) {
    write(2, errstr, strlen(errstr));
```

Месяц назад Мы познакомились с демонами по Асприну.

ВВОД/ВЫВОД

```
    return EXIT_FAILURE;
  }
  printf ("Hello!\n");
  return EXIT_SUCCESS;
}
```

Функция isatty(3) позволяет программе узнать, является ли предоставленное ей устройство ввода/вывода терминалом. Аргументом isatty() должен быть дескриптор файла устройства, который мы получаем из переменной stdout, пользуясь функцией fileno(3). Функция isatty() возвращает единицу, если переданный ей дескриптор соответствует терминалу, и 0 – в противном случае. Программа noredirect напечатает строку "Hello!", только если у нее есть доступ к терминалу. Если же вы скомандуете

```
noredirect > file
```

на экране появится гневное сообщение, а файл **file** окажется пустым. Впрочем, возможно, вы не всегда будете столь строгим по отношению к пользователю. Вам может потребоваться частично запретить перенаправление для выделенных мест программы. В этом случае можно воспользоваться трюком, который показан в программе *noredirect2* (**noredirect2.c** на диске).

```
#include <stdio.h>
#include <stdlib.h>
#include <fcntl.h>
#include <string.h>

int main (int argc, char ** argv)
{
  int fd_in, fd_out;
  char * nr_message = "Enter your name, please\n";
  char buf[255];
  fd_in = open("/dev/tty", O_RDONLY);
  fd_out = open("/dev/tty", O_WRONLY);
  write(fd_out, nr_message, strlen(nr_message));
  read(fd_in, buf, 255);
  printf("Your name is %s\n", buf);
  return EXIT_SUCCESS;
}
```

В этой программе мы напрямую обращаемся к устройству, которое соответствует управляющему терминалу ввода/вывода. Вы, конечно, помните, что устройства в Unix/Linux представлены файлами. Файл **/dev/tty** предоставляет программе доступ к ее управляющему терминалу. Иначе говоря, для каждой программы файл **/dev/tty** представляет тот терминал, который является управляющим терминалом данной программы. Мы открываем устройство **/dev/tty** дважды – для чтения и записи. Запись в файл **/dev/tty** соответствует выводу данных на экран терминала, а чтение – вводу данных с клавиатуры. Полученные дескрипторы, fd_in и fd_out, мы будем использовать, соответственно, для ввода/вывода, который нельзя перенаправить. Вся магия перенаправления ввода/вывода основана на том, что программы пользуются потоками ввода-вывода, которые предоставляет им оболочка. Если мы в программе свяжем дескрипторы с конкретным устройством, команды оболочки уже не смогут этого изменить. Таким образом, строка "Enter your name, please" всегда будет выводиться на терминал. С терминала же программа будет считывать ответ пользователя. В то же время, строка, распечатанная с помощью printf(), может быть перенаправлена на другое устройство, поскольку printf() использует поток вывода, предоставленный оболочкой (мы можем, в принципе, подменить поток, используемый printf() по умолчанию, но не будем этого делать).

Управление терминалом

Управление терминалом осуществляется с помощью структуры termios, которая содержит значения и флаги, влияющие на различные параметры терминала. Прежде чем изучать структуру termios и работающие с нею функции, необходимо сказать несколько слов о режимах работы терминала. Поведение терминала во многом определяется тем, в каком режиме он находится – каноническом или неканоническом. Примером программы, использующей терминал в каноническом режиме, может служить оболочка *bash*. В каноническом режиме терминал передает программе символы, введенные пользователем, только после того, как пользователь нажмет клавишу Ввод. До тех пор, пока пользователь не нажал Ввод, он может редактировать вводимую строку, используя клавиши BackSpace, Del, Tab и им подобные. Если терминал находится в неканоническом режиме, пользователю не требуется нажимать Ввод для того, чтобы программа получила напечатанные им символы. В неканоническом режиме программа может получать символы сразу после ввода или с задержкой, по одному или по нескольку, в зависимости от настроек режима терминала. Поскольку в неканоническом режиме все введенные символы сразу же передаются программе, возможность редактировать строку у пользователя отсутствует. Независимо от режима, в котором находится терминал, можно установить некоторые базовые параметры, например, скорость передачи данных и функцию контроля ошибок (очень полезно при модемном соединении). Впрочем, некоторые из них, например отключение отображения вводимых пользователем символов (echo), могут с пользой применяться и в современных программах.

Структура termios позволяет управлять флагами и численными параметрами, которые можно разделить на пять групп: ввод, вывод, управление оборудованием, локальные параметры и специальные управляющие символы. Простейшая структура termios состоит из пяти полей, соответствующих перечисленным группам:

```
struct termios {
// флаги управления вводом
tcflag_t  c_iflag;
// флаги управления выводом
tcflag_t  c_oflag;
// флаги управления оборудованием
tcflag_t  c_cflag;
// флаги управления локальными параметрами
tcflag_t  c_lflag;
// Специальные управляющие символы
cc_t c_cc[NCCS]
};
```

У структуры termios могут быть и другие поля, но нас они не интересуют. Обычно работа со структурой termios происходит по следующему сценарию (все необходимые функции и типы данных определены в файле **termios.h**): с помощью функции tcgetattr(3) мы получаем копию

структуры, описывающую текущее состояние терминала, и делаем еще одну копию. Затем мы модифицируем значения полей одной из копий termios так, чтобы изменить нужные нам параметры терминала, и передаем драйверу терминала новое значение termios с помощью функции tcsetattr(3). После того, как работа с терминалом в нестандартном режиме закончена, мы восстанавливаем исходное состояние терминала с помощью сохраненной копии исходной структуры termios и функции tcsetattr(). Первым аргументом функции tcgetattr() должен быть дескриптор файла, соответствующего терминалу. Вторым аргументом является указатель на структуру termios, в которой функция возвращает текущие настройки терминала. Первым параметром функции tcsetattr() также служит дескриптор файла терминала. Второй параметр используется для передачи флагов, определяющих, когда изменения параметров терминала должны вступить в силу. Третьим параметром tcsetattr() является указатель на структуру termios, содержащую новые параметры.

Ключевой момент во всем этом – модификация полей структуры termios. Первые четыре поля структуры содержат комбинации флагов, определяющих параметры терминала. Пятое поле представляет собой массив значений. Индексам этого массива соответствуют специальные константы, с помощью которых мы можем понять значение элементов массива. Рассмотрим сначала поля termios, содержащие флаги. Полное описание флагов (а их довольно много) можно найти на странице man, посвященной termios. Я перечислю здесь только некоторые флаги, которые устанавливаются в поле c_lflag, поскольку они представляются мне наиболее интересными. Флаг ECHO управляет отображением вводимых символов на экране монитора. Если он установлен, символы отображаются, в противном случае – нет. Флаг ECHOE делает то же, что флаг ECHO, но только для управляющих символов, стирающих другие символы или строки (например, BackSpace). Поскольку неканонический режим не поддерживает редактирование строки, в этом режиме флаг ECHOE игнорируется. Если установлен ICANON, терминал находится в каноническом режиме, в противном случае – в неканоническом. Флаг IEXTEN переводит терминал в режим расширенной обработки вводимых символов. От того, установлен ли флаг ISIG, зависит, будут ли специальные комбинации клавиш, такие как Ctrl-C и Ctrl-Z, инициировать соответствующие им сигналы.

Из констант, соответствующих индексам массива c_cc[], наибольший интерес представляют две – VMIN и VTIME. Чтобы объяснить важность этих параметров, рассмотрим подробнее работу терминала в неканоническом режиме. В каноническом режиме сигналом завершения ввода данных является нажатие клавиши Ввод. Любая функция, считывающая данные с терминала, вернет управление вызвавшей ее программе только после того, как пользователь нажмет эту клавишу. При работе терминала в неканоническом режиме дело обстоит сложнее. В этом режиме нет сигнала, который бы оповещал систему о том, что ввод данных окончен и функция чтения данных должны вернуть управление. Поведение функций, читающих данные в этом режиме, зависит от параметров termios.c_cc[VMIN] и termios.c_cc[VTIME]. Параметр termios.c_cc[VMIN] указывает минимальное число введенных символов, после которого функция, считывающая данные, может вернуть управление программе. Параметр termios.c_cc[VTIME] указывает максимальное время ожидания ввода (после ввода первого символа), по прошествии которого функция чтения данных возвращает управление, независимо от того, сколько символов было прочитано. Если обоим этим параметрам присвоены значения, большие нуля, функция чтения данных вернет управление после того, как будет выполнено требование одного из параметров. Если функция получит количество символов, заданное в параметре termios.c_cc[VMIN] до истечения срока времени, заданного в параметре termios.c_cc[VTIME], она вернет управление и передаст программе соответствующее количество символов. Если заданный промежуток времени истечет до того, как функция считает указанное ей количество символов, функция вернет управление и передаст программе те символы, которые она успела считать. При этом, как следует из сказанного выше, функция чтения данных вернет как минимум один символ. Если одному из параметров termios.c_cc[VMIN] или termios.c_cc[VTIME] присвоено нулевое значение, а другому – ненулевое, то условием возврата из функции чтения данных становится значение ненулевого параметра. Наконец, если оба параметра имеют нулевые значения, функция чтения данных всегда будет возвращать управление немедленно. Если к моменту вызова функции чтения данных в потоке ввода были символы, буфер функции будет заполнен ими, иначе функция чтения данных вернет пустой буфер.

Вернемся к функции tcsetattr(). Во втором параметре этой функции может быть передан один или несколько нижеследующих флагов (для комбинации флагов используется оператор |).

- **TCSANOW** – изменения параметров терминала вступают в силу немедленно.
- **TCSADRAIN** – изменения параметров терминала вступают в силу после того, как все данные, записанные ранее в файл устройства, будут переданы самому устройству. Это значение обычно используется, если новые значения полей termios изменяют параметры вывода данных.
- **TCSAFLUSH** – изменения параметров терминала вступают в силу после того, как все данные, записанные ранее в файл устройства, будут переданы самому устройству. Все данные, которые в этот момент были введены с клавиатуры, но еще не прочитаны программой, при этом теряются.
- **TCSASOFT** – этот флаг заставляет функцию игнорировать значения полей c_cflag, c_ispeed и c_ospeed структуры termios. Поля c_ispeed и c_ospeed управляют скоростью передачи входящих и исходящих данных. Поскольку мы не рассматриваем подключение терминала с помощью модема, эти поля нас не интересуют.

Настало время освежить теорию практикой. Рассмотрим два примера, в которых изменение свойств терминала может оказаться полезным. Первый из них – это консольная программа, предназначенная для ввода пароля. Как вы знаете, во время ввода пароля такие программы не отображают никаких символов. Теперь вы также догадываетесь, как они это делают – им достаточно передать системе структуру termois со сброшенным флагом ECHO. Программа *passwdmode* (файл **passwdmode.c**) демонстрирует эту технику:

```
#include <stdio.h>
#include <stdlib.h>
#include <signal.h>
#include <termios.h>

#define BUF_SIZE 15

int main (int argc, char ** argv)
{
  struct termios oldsettings, newsettings;
  char password[BUF_SIZE+1];
  int len;
  sigset_t newsigset, oldsigset;
  sigemptyset(&newsigset);
  sigaddset(&newsigset, SIGINT);
  sigaddset(&newsigset, SIGTSTP);
  sigprocmask(SIG_BLOCK, &newsigset, &oldsigset);
  tcgetattr(fileno(stdin), &oldsettings);
  newsettings = oldsettings;
  newsettings.c_lflag &= ~ECHO;
  tcsetattr(fileno(stdin), TCSAFLUSH, &newsettings);
  printf("Enter password and press [Enter]\n");
  len = read(fileno(stdin), password, BUF_SIZE);
  password[len] = 0;
  tcsetattr(fileno(stdin), TCSANOW, &oldsettings);
  sigprocmask(SIG_SETMASK, &oldsigset, NULL);
  printf("Your password is %s\n", password);
  return EXIT_SUCCESS;
}
```

В начале программы мы блокируем сигналы SIGINT и SIGTSTP (зачем это нужно, я объясню ниже). Затем с помощью функции

tcgetattr() мы заполняем переменную oldsettings типа struct termios текущими значениями параметров терминала. Далее мы копируем содержимое oldsettings в переменную newsettings. Строка

```
newsettings.c_lflag &= ~ECHO;
```

сбрасывает флаг ECHO в структуре newsettings. Остальные параметры терминала остаются без изменений. Далее, с помощью функции tcsetattr() мы устанавливаем новые параметры. Теперь терминал не будет выводить на экран символы, вводимые пользователем, и мы можем вызвать функцию, считывающую значение пароля. После этого программа восстанавливает прежнее состояние терминала. После этого можно разблокировать заблокированные сигналы. Мы распечатываем строку с введенным «паролем» (не вздумайте вводить в программе какой-нибудь настоящий пароль, иначе злоумышленник, прячущийся за вашей спиной, обязательно его увидит). Зачем мы блокировали сигналы во время ввода пароля? Представьте себе, что в то время, когда программа ожидает ввода пароля, пользователь передумал и захотел завершить ее с помощью Ctrl-C. Если программа завершится в этот момент, состояние терминала не будет восстановлено, и символы, вводимые пользователем, по-прежнему не будут отображаться. Это не смертельно, но неудобно. Вот почему программы, ожидающие ввода пароля, временно блокируют некоторые сигналы.

Рассмотрим теперь другой случай. Представьте себе, что программа выполняет некую длительную операцию, и вы хотите, чтобы у пользователя была возможность прервать ее, не прибегая к таким средствам, как сигналы. Для этого программа может периодически проверять, не нажал ли пользователь клавишу выхода, например q. Однако если терминал находится в каноническом режиме, пользователю придется нажать еще и «Ввод», чтобы программа могла считать символ, и это не говоря о том, что в каноническом режиме функции чтения данных по умолчанию блокируют выполнение программы до появления данных. Нам нужно, чтобы функция ввода, например, getchar(), проверяла наличие в потоке ввода символа q, причем без всякого дополнительного символа ввода, но возвращала управление немедленно, независимо от того, есть символ в потоке ввода или нет. Все это очень просто сделать, переведя терминал в неканонический режим, что демонстрирует программа *pressq* (файл **pressq.c**)

```
#include <stdio.h>
#include <stdlib.h>
#include <termios.h>

int main (int argc, char ** argv)
{
  struct termios oldsettings, newsettings;
  tcgetattr(fileno(stdin), &oldsettings);
  newsettings = oldsettings;
  newsettings.c_lflag &= ~(ECHO|ICANON|ISIG);
  newsettings.c_cc[VMIN] = 0;
  newsettings.c_cc[VTIME] = 0;
  tcsetattr(fileno(stdin), TCSANOW, &newsettings);
  while(getchar() != 'q') {
    sleep(1);
    printf("press [q] to quit\n");
  }
  tcsetattr(fileno(stdin), TCSANOW, &oldsettings);
  return EXIT_SUCCESS;
}
```

Мы получаем структуру с текущими параметрами терминала, и сохраняем ее, как и в предыдущем случае. Затем мы сбрасываем сразу три флага: ECHO, ICANON и ISIG. Что дает сброс флага ECHO, вы уже знаете. Мы не хотим, чтобы введенный пользователем символ команды отображался на экране. Сброс флага ICANON переводит монитор в неканонический режим, а сброс флага ISIG приводит к тому, что ввод специальных символов Ctrl-C и Ctrl-Z не порождает сигналов. Это еще один способ защитить программу от некорректного завершения в тот момент, когда параметры терминала изменены. Затем мы устанавливаем значения параметров newsettings.c_cc[VMIN] и newsettings.c_cc[VTIME]. Обоим параметрам присваивается значение 0. В результате функция getchar() всегда будет возвращать управление немедленно. Далее программа переходит в бесконечный цикл, из которого ее может вывести только появление символа q в потоке ввода. По выходе из цикла мы восстанавливаем параметры терминала и завершаем программу. Необходимо подчеркнуть разницу между блокированием сигналов в программе *psswdmode* и сбросом флага ISIG в программе *pressq*. В первом случае обработка сигналов откладывается. Если во время ввода пароля пользователь нажмет Ctrl-C, программа получит соответствующий сигнал после того, как пользователь нажмет ввод (и после того, как программа вернет терминал в нормальный режим). Если же вы сбрасываете флаг ISIG, специальные сочетания клавиш не будут инициировать сигналы и программа вообще их не получит.

Тема консольного ввода/вывода неисчерпаема, но мы не собираемся на ней долго задерживаться. В следующей статье, последней в этой серии, мы рассмотрим интерфейс terminfo и библиотеку *ncurses*. LXF

» **Через месяц** Финальная серия: мы изучим ncurses и terminfo!

Телефонная книга:

ЧАСТЬ 2 Встречают по одежке – и Большой Босс не был сильно впечатлен созданной нами в прошлый раз адресной книгой. **Александр Бабаев** исправляет замеченные недочеты.

Наш эксперт

Александр Бабаев
Разработчик открытой мультиблог-системы jDnevnik – победитель конкурсов IBM WAS CE Contest 2006 и конкурса проектов для разработчиков на Java – Java конкурс 2005.

В прошлый раз мы создали простейшую электронную записную книжку. Она работает в браузере и показывает несколько простых страничек, на которых можно просмотреть список контактов, добавить новый контакт, удалить его или отредактировать. А сейчас давайте попробуем сделать все это более правильно.

Почему было плохо?

Действительно, почему? Работает, и хорошо. Достаточно быстро и не слишком сложно. Но вдруг захочется поменять дизайн страничек? А захочется через десять минут работы. Или после того, как страничку посмотрит начальник.

Чтобы сделать это, можно изменить код проекта, потом перекомпилировать его, остановить сервер (А? Кто-то работал? Извините...), установить новый код и повторно запустить сервер. Метод, мягко говоря, неудобный. А можно изменить сам проект так, чтобы выполнение таких пожеланий не требовало столь сложных действий. Второй путь зовется рефакторингом и гораздо более корректен. Если разделить дизайн и логику работы приложения (бизнес-логику), то в дальнейшем можно будет, например, разделить и работу по их поддержанию. Хороший программист не всегда создает хорошие пользовательские интерфейсы, поэтому данный аспект тоже важен.

Как сделать хорошо?

Ну, вкратце уже понятно. Нужно вынести в отдельные файлы ту часть, которая меняется часто (в нашем случае, это интерфейс) и как-то подключить эти файлы из нашей программы. Плюс, желательно сделать это так, чтобы формат файлов «дизайна» был стандартным, чтобы каждый раз не переучиваться.

Решений для данной проблемы существует множество. Рассмотрим самые распространенные:

» **Шаблоны**. Одна из самых распространенных библиотек работы с шаблонами – *Velocity*. При использовании шаблонных движков можно добавлять в текст специальные вставки, которые говорят: «Тут вставить значение переменной Name». Иногда можно делать более сложные операции (вставка подшаблонов, вычисления, условные вставки).

» **JSP (Java Server Pages)**. По времени появления, пожалуй, первая технология для отделения дизайна от бизнес-логики. Но я ее поставил второй, так как она сложнее, чем просто шаблонная библиотека. JSP позволяет внедрить код на (по задумке) любом языке программирования внутрь специальным образом созданной странички. Впрочем, обычно используется Java. Теоретически, можно написать серверное приложение, используя исключительно JSP. Этот подход похож на PHP, с тем отличием, что JSP-страницы – это полноценные сервлеты, они компилируются при обновлении исходного текста и обрабатываются как таковые.

» **JSF (Java Server Faces)**. В некотором роде эта технология объединяет подходы, которые используются при создании «обычных» и «сетевых» программ. Интерфейс (как дизайн интерфейса, так и его логика) программы описывается специальным образом, а после этого пишутся JSP-странички, в которых указывается «тут вставить таблицу с именем таким-то». JSF обрабатывает эти спецвставки и «рисует» функциональные элементы интерфейса (обрабатывая события от них и так далее), позволяя дизайнеру сосредоточиться на остальном.

» **Google Web Toolkit**. Не могу не остановиться на этом средстве. При его использовании на выходе получается полноценное AJAX-приложение (что это такое – тема отдельной статьи, пример – Google Mail), а на входе – все тот же Java-код. Решение интересное, не лишенное своих достоинств и недостатков.

Мы же в рамках данной статьи рассмотрим «средненькое» решение – Java Server Pages. В основном – из-за его стандартности, хотя для данного конкретного случая можно выбрать какой-нибудь шаблонный движок, например, тот же Velocity (http://velocity.apache.org).

Общая схема работы приложения

Поняв, что нужно отделить логику от дизайна, давайте подумаем, каким образом это можно сделать. Предлагаю остановиться на следующей схеме (**Рис. 1**):

Рис. 1. Схема работы сервлета при разделении логики и дизайна.

Сервлет выдает данные, абсолютно не заботясь о том, как они будут отображаться. Но выдает он их не в «сыром» виде, а в полностью обработанном, готовом для отображения на экране (например, если нужно полное имя человека, а в данных – его ФИО по отдельности, то сервлет должен преобразовать второе в первое перед передачей в JSP).

Возникает вопрос: как же передаются данные от сервлета в JSP? Через уже известный нам объект request. К нему «прикручен» специальный ассоциативный массив «String – Object», который называется атрибутами и который живет, пока жив запрос. К нему имеет доступ и

переход на JSP

сервлет, и JSP-страница, поэтому его можно (и это правильно) использовать для передачи данных.

Переходим на Tomcat

Но сначала нужно переписать наш сервлет «по-взрослому». Встроенный сервер – это замечательно для кустарных проектов, но обычно контейнер сервлетов уже стоит, и подключаться следует к нему.

Мы будем использовать Tomcat 5.5. Это классический, можно даже сказать, стандартный открытый сервлет-контейнер. Для установки Tomcat достаточно просто скачать его с http://tomcat.apache.org (или взять с нашего DVD), распаковать и запустить bin/startup.sh (или соответсвующий **.bat**). Tomcat работает с файлами специального типа Web Archive (WAR). Обнаружив такой файл в определенном каталоге, Tomcat разворачивает его и запускает содержащееся в нем приложение. Чтобы перезапустить или обновить программу, достаточно просто заменить один WAR-файл другим.

Предыдущий код не готов для работы с Tomcat, поэтому его нужно немного переписать. Вот что будет сделано:

» AddressBook потеряет методы start и main и превратится в простое хранилище записей.

» AddressBookHandler превратится в AddressBookServlet, и в него будет добавлено примерно следующее (**Листинг 1**):

Листинг 1. Новый AddressBook

```
private AddressBook _addressBook = null;

public void init(ServletConfig aServletConfig) throws ServletException {
 super.init(aServletConfig);
 _addressBook = new AddressBook();
}

protected void doGet(HttpServletRequest aRequest, HttpServletResponse
aResponse)
      throws ServletException, IOException
 handle(aRequest, aResponse);
}

protected void doPost(HttpServletRequest aRequest, HttpServletResponse
aResponse)
      throws ServletException, IOException
 handle(aRequest, aResponse);
}
```

» Сам метод handle тоже слегка преобразуется (**Листинг 2**):

Листинг 2. Новый метод handle

```
private void handle(HttpServletRequest aRequest, HttpServletResponse
aResponse)
      throws ServletException, IOException {
 aRequest.setCharacterEncoding("utf-8");

 String target = aRequest.getRequestURI().substring(
  aRequest.getContextPath().length());

 if (target.equals("/")) {
 _drawer.outputPage("index.jsp", aRequest, aResponse);
 } else if ("/add".equals(target)) {
 handleAdd(aRequest, aResponse);
 } else if ("/view".equals(target)) {
 handleView(aRequest, aResponse);
 } else if ("/edit".equals(target)) {
 handleEdit(aRequest, aResponse);
 } else if ("/remove".equals(target)) {
 handleRemove(aRequest, aResponse);
 }
}
```

» Для того, чтобы Tomcat «понял», что ему положили сервлет, и знал, как его обрабатывать, нужно написать специальный файл, который называется «дескриптор». Несмотря на то, что слово страшное, это просто XML-документ с описанием сервлета. Если перевести с языка написания дескрипторов на русский, то получится примерно следующая информация:

- Наш сервлет называется «ABServlet» и запускается классом AddressBookServlet. Теоретически можно назвать сервлет так же, как и класс, но мы не будем так делать, чтобы было меньше путаницы.
- Для всех URL, которые начинаются с «/», нужно вызывать сервлет, который называется ABServlet.

А вот как он выглядит (**Листинг 3**):

Листинг 3. Дескриптор для сервлета

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.4"
  xmlns="http://java.sun.com/xml/ns/j2ee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
   http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd" >

 <servlet>
 <display-name>AddressBook</display-name>
```

```
<servlet-name>Servlet</servlet-name>
<servlet-class>AddressBookServlet</servlet-class>
<load-on-startup>0</load-on-startup>
</servlet>

<servlet-mapping>
<servlet-name>Servlet</servlet-name>
<url-pattern>/</url-pattern>
</servlet-mapping>
</web-app>
```

» Дескриптор будет называться **web.xml** и храниться в специальном каталоге. Где именно – обсудим, когда будем собирать сервлет в WAR.

Сделайте указанные изменения самостоятельно или возьмите готовый код с DVD. Все в порядке? Тогда движемся дальше.

Новый метод

Если присмотреться более внимательно к коду нового handle, можно заметить, что там появился вызов метода outputPage. Раньше его, в отличие от разных handle… не было. Это метод, который выбирает JSP-файл и передает ему управление для вывода страничек. Выглядит метод следующим образом (**Листинг 4**):

Листинг 4. Метод outputPage

```
public void outputPage(String aJSPName, HttpServletRequest aRequest,
  HttpServletResponse aResponse) throws IOException, ServletException
{
 RequestDispatcher dispatcher = aRequest.getRequestDispatcher("/jsps/"
+ aJSPName);
 dispatcher.forward(aRequest, aResponse);
}
```

В этом методе мы берем нужный JSP-файл и говорим сервлет-контейнеру: «Обработай, пожалуйста». Остальное берет на себя контейнер. Он ищет JSP-файл, загружает его, компилирует (если это нужно), выполняет получившийся сервлет, а результат записывает в aResponse.

JSP-страницы

Для начала создадим каталог, в котором будем собирать наше интернет-приложение. Назвать можно как угодно, например, WebApp (Web Application). В нем создадим специальный каталог **WEB-INF**, где должен находиться дескриптор **web.xml**, и каталог **jsps**, в котором будут храниться JSP-странички.

Создадим три JSP-файла: для индексной странички, для редактирования (или добавления) записей и для просмотра, и назовем их, соответственно, **index.jsp**, **edit.jsp**, **view.jsp**. Не забудьте – их нужно сохранить в в WebApp/jsps.

Сам JSP достаточно прост. Рассмотрим **index.jsp** (**Листинг 5**):

Листинг 5. index.jsp

```
<%@ page contentType="text/html; charset=UTF-8" %>
<html>
 <head>
 <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
 <title>Адресная книга</title>
 </head>

 <body><h1>Адресная книга</h1>
 <a href="<%=request.getContextPath()%>/add">Добавить запись</
a><br/>
 <a href="<%=request.getContextPath()%>/view">
Просмотреть записи</a><br/>
 </body>
</html>
```

Первая строчка добавляет поле «Content-type» к HTTP-заголовку ответа. Это прямой аналог строки

```
aRequest.setContentType("text/html; charset=utf-8")
```

из «старого» метода handle. А дальше, кроме странных вставок <%…%>, идет обычный HTML-код. И это хорошо! Это понятно! Теперь разберемся с непонятным.

В JSP можно вставлять «инородный» для HTML код, который специальным образом интерпретируется сервером и может быть использован для вставки различных данных. Есть несколько типов таких вставок.

» <%@…%> – обозначает специальную вставку, которая определяет параметры страницы, в нашем случае – ContentType. Можно задавать, например, язык, на котором написана страница. Он же используется для секций import (см. **view.jsp** ниже).

» <%=…%> – это простой вывод переменной. Действие вставки <%=что-нибудь%> аналогично вызову request.getWriter().write(что-нибудь).

» <%…%> – самый общий вариант вставки, внутри может быть любой код. В нашем случае, на Java.

index.jsp – простой файл, посмотрим на нечто более сложное. Например, **view.jsp** (**Листинг 6**).

Листинг 6. view.jsp

```
<%@ page contentType="text/html; charset=UTF-8" %>
<%@ page import="java.util.*" %>
<html>
<head><title>Адресная книга</title></head>

<body><h1>Адресная книга, список контактов</h1>
 <a href="<%=request.getContextPath()%>">На главную</a><br/>

 <span style="color: green;"><%=request.getAttribute("message")%></
span>

 <table border="1">
 <tr><td width="100">Имя</td><td width="100">Номер</td><td
width="100">Комментарий</td><td> - </td></tr>
 <% Map numbers = (Map) request.getAttribute("numbers");
  Map comments = (Map) request.getAttribute("comments");

  for (Object entry : numbers.entrySet()) {
  String name = (String) ((Map.Entry) entry).getKey();
  String number = (String) numbers.get(name);
  String comment = (String) comments.get(name); %>
  <tr>
   <td class="name"><%=name%></td>
   <td class="number"><%=number%></td>
   <td class="comment"><%=comment%></td>
   <td class="name">
<a href="<%=request.getContextPath()%>/remove?number=<%=number%
>">Удалить</a>
<a href="<%=request.getContextPath()%>/edit?number=<%=number%>">
Редактировать</a>
   </td>
  </tr>
 <% } %>
 </table>
</body>
</html>
```

Как можно заметить, здесь есть и импорт (о чем я говорил чуть выше), и вставка Java-кода. Данный файл отлично показывает, как, например (не самый лучший способ, конечно), сделать вывод в цикле.

А как это обрабатывается-то?

Естественно, и методы handle… после такого изменения стали другими. Весь вывод HTML-кода исчез, осталась подготовка данных, и вызов метода outputPage. Вот, например, метод handleEdit(…) (**Листинг 7**):

Листинг 7. Метод handleEdit, обработка редактирования записи

```
if (aRequest.getParameter("number") == null) {
 _addressBook.removeContactByNumber(aRequest.
getParameter("number"));
 aRequest.setAttribute("message", "Не определено, что
редактировать");
 handleView(aRequest, aResponse);
} else if (aRequest.getParameter("edited") != null) {
 _addressBook.editContact(aRequest.getParameter("edited"),
  aRequest.getParameter("name"),
  aRequest.getParameter("number"),
  aRequest.getParameter("comment"));
 aRequest.setAttribute("message", "Контакт \"" +
    aRequest.getParameter("name") + "\" отредактирован");
 handleView(aRequest, aResponse);
} else {
 Contact contact = _addressBook.getContactByNumber(aRequest.
getParameter("number"));
 aRequest.setAttribute("action", "edit");
 aRequest.setAttribute("edit.name", contact.getName());
 aRequest.setAttribute("edit.number", contact.getNumber());
 aRequest.setAttribute("edit.comment", contact.getComment());
 outputPage("edit.jsp", aRequest, aResponse);
}
```

Остальные методы меняются аналогично – их полный код можно найти на диске.

И как все это вставить в Tomcat?

Теперь у нас есть:

- Классы Contact, AddressBook, AddressBookServlet.
- Файл **web.xml**.
- Каталог **jsps** с файлами **edit.jsp**, **index.jsp**, **view.jsp**.

Для того, чтобы Tomcat понял, что ему дали полноценное приложение, нужно выполнить всего три шага:

- Скомпилировать все, что компилируется, и создать правильную иерархию файлов и каталогов, которая представлена на **рис. 2**.

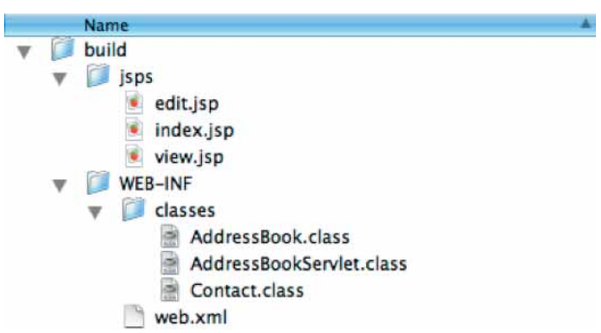


Рис. 2. Структура файлов и каталогов, которую мы будем собирать в war.

- Создать специальный файл-описание архива («манифест»).
- Заархивировать созданную структуру при помощи утилиты *jar*, входящей в комплект JDK.

Скомпилируем файлы. Тут ничего нового не появилось, разве что изменилась сама команда (обратите внимание на ключ -cp, задающий библиотеки classpath):

```
cd ~/Programming/AddressBook/src
javac -encoding utf-8 -cp ~/bin/tomcat/common/lib/servlet-api.jar
    -d ../build/WEB-INF/classes/ *.java
```

Переходим к созданию манифеста. Он должен называться **MANIFEST.MF** и располагаться в каталоге **META-INF**. К счастью, за этим следит сам **jar**, поэтому нам достаточно просто сохранить где-то файл и указать его **jar**'у как манифест. В нашем случае он предельно прост и не содержит интересной информации, но в принципе здесь могут располагаться всякие настройки для запуска вашего приложения. Вот его текст (**Листинг 8**):

Листинг 8. Манифест для war-файла

```
Manifest-Version: 1.0
Created-By: Hands of programmer
```

Теперь соберем все в war (Web Archive). Манифест для приведенной ниже команды должен быть назван **MANIFEST.MF** и располагаться рядом с каталогом **build**. Результирующий архив называется **address.war** и располагается там же, рядом с манифестом.

```
jar -cfm ../address.war ../MANIFEST.MF *
```

А сейчас наступает самый волшебный момент! Возьмите **address.war** и положите его в каталог webapps Tomcat'а. Подождите несколько секунд. Увидев новое приложение, Tomcat развернет его (появляется каталог с именем вашего war'а) и подключит к системе. После этого можно просто зайти в браузер и набрать:

http://localhost:8080/address/

Вуаля, получите ваше приложение.

И что теперь?

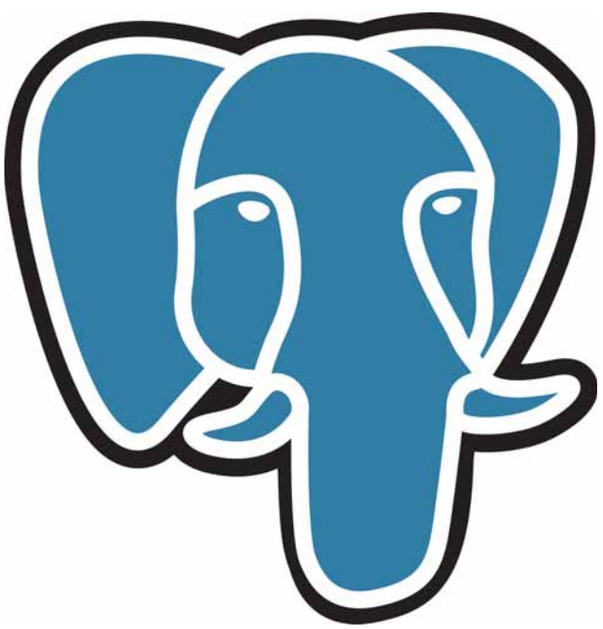
А теперь можно менять JSP-файлы «на лету» в распакованном каталоге **webapps/address/jsps**. При этом будет автоматически происходить несколько действий, в результате которых файлы подхватятся приложением. Так меняется дизайн без перекомпиляции, без рестарта серверного приложения, как это у нас было до сих пор.

Я считаю, что на данном этапе приложение «Адресная книга» работает хорошо. Оно выполняет свои несложные функции и умеет изменяться «на лету» по запросу пользователя. Оно простое – и это чуть ли не самое главное. Но есть еще достаточно аспектов, о которых стоит знать при разработке более сложных интернет-приложений. Мы рассмотрим их в следующих статьях данной серии. LXF

Через месяц Паттерн MVC как основа для построения Web-приложений.

Настройка

ЧАСТЬ 5 PostgreSQL в фундаменте «1С:Предприятия»? В свое время эта новость произвела фурор в мире Linux, а сегодня **Евгений Балдин** покажет, что для этого было сделано, и расскажет, как настроить PostgreSQL для выполнения этих или любых других задач.



– Между прочим, – сказал Роман громко, – уже в течение двух минут я пытаюсь его пассивизировать, и совершенно безрезультатно.

А. и Б. Стругацкие «Понедельник начинается в субботу»

Тюнинг – это не операция: это образ жизни. Очевидно [по крайней мере, автору, – прим. ред.], что когда необходимые характеристики можно улучшить несколькими способами, то всеми этими способами их улучшать и надо. Опять же, следует помнить, что избыточная и ранняя оптимизация – корень многих зол. Если система работает и «не кашляет», то, может быть, не стоит ее «настраивать»?

Об оборудовании

«Театр начинается с вешалки», а большая база данных начинается с выбора сервера. *PostgreSQL* умудряется работать почти при любой конфигурации, но если вас интересует результат, то следует знать, на что обращать внимание. Понятно, что ограничения на оборудование диктуются, в основном, объемом денежных ресурсов, но в имеющихся пределах всегда можно что-то подвигать.

Как и для всякой программы, работающей с большим объемом данных, дисковая подсистема является для *PostgreSQL* определяющей. Поэтому, если есть возможность, то лучше вложиться именно в нее. В противоположность дисковой подсистеме процессор нагружается не очень сильно. Поэтому для сервера достаточно, чтобы процессор просто был, хотя лучше, чтобы он был не один. Если на покупку хорошего SCSI-диска денег нет, то следует вложиться в память.

По поводу дисков можно сказать, что чем их больше – тем лучше. По возможности, следует выделить отдельный диск для журнала транзакций (**pg_xlog**). Избыток дисков так же позволит собрать из них RAID. Никто не отменяет резервирование, но дополнительная избыточность для дисковой подсистемы, равно как и источники бесперебойного питания, сэкономят массу сил и нервов.

Относительно недорогие системы снабжены дешевыми встроенными RAID-контроллерами на 4 диска. Не следует использовать эти контроллеры, лучше настроить программный RAID. Не надо использовать RAID 5[1] для небольшого числа дисков. До 6 дисков включительно лучше оформить как RAID 1+0[2]. Избыточность во всем – это похоже на паранойю.

На сервер, где уже работает *PostgreSQL*, не следует «подселять» другие базы данных или программы, осуществляющие интенсивный обмен с дисковой памятью. А вот программы, которые вместо этого интенсивно используют процессор, например, Apache, вполне уживаются, если памяти достаточно.

Конфигурационные файлы

Конфигурационных файлы – не единственный способ настройки сервера БД. Умолчания можно изменить непосредственно при сборке из исходников. Значения можно передавать непосредственно серверу postmaster в командной строке, используя ключ -c. Также можно определить переменную окружения PGOPTIONS, а значения некоторых переменных можно изменить прямо в процессе выполнения запросов.

Но работать с конфигурационными файлами все же придется. По умолчанию *PostgreSQL* настроен на минимальное потребление ресурсов, и это не может не сказаться на скорости. То, что подходит для локальной записной книжки, не годится для боевого сервера.

Все настройки очень подробно описаны в документации. Для любителей «выжимать из программы все» существует специальный список рассылки psql-performance: **http://archives.postgresql.org/pgsql-performance/**. В документации на странице Power PostgreSQL **http://www.powerpostgresql.com/Docs** также можно найти некоторое количество полезных подсказок.

Для настройки *PostgreSQL* используются следующие файлы:

[1] Один диск в массиве выделяется под контрольные суммы
[2] зеркалирование (1) + объединение (0)

Месяц назад Интерфейсы PostgreSQL: но на сей раз не пользовательские, а программные.

PostgreSQL

» **pg_hba.conf** – политика доступа и идентификации пользователей,
» **postgresql.conf** – собственно, настройки сервера.

pg_hba.conf

При использовании *PostgreSQL* в качестве «личной записной книжки» зачастую не требуется открывать сетевой доступ к базе данных. По умолчанию *PostgreSQL* настроен так, что каждый локальный пользователь может подсоединиться к одноименной базе данных, при условии, что таковая уже создана.

Но это не значит, что так бывает всегда. *PostgreSQL* предоставляет свои механизмы для управления пользователями с помощью тройки команд CREATE USER, DROP USER и ALTER USER. В случае несовпадения регистрационных имен в *PostgreSQL* и в системе или при работе с БД с других компьютеров необходимо «обговорить» правила доступа к данным на сервере.

Для настройки политики доступа к серверу волей-неволей придется заглянуть в файл **pg_hba.conf**[3]. Файл разбит на строки, каждая из которых регулирует правила получения доступа для конкретной машины или для целой группы IP. Общий вид **pg_hba.conf** примерно таков:

```
# Разрешаем доступ через локальные unix-сокеты абсолютно
#всем пользователям к базам данным, совпадающим по названию
#с регистрационными именами
local   all         all                           ident sameuser
# Доверяем пользователю alex с указанного IP безгранично
#в рамках базы данных photos
host    photos      alex   130.255.204.48/32  trust
# Требуем пароль от пользователя baldin при доступе с
#компьютеров из сети 128.138.242.192/27 к базам данных
#data и photos
hostssl data,photos  baldin 128.138.242.192/27  md5
```

Как можно догадаться (или подсмотреть в комментариях к исходному **pg_hba.conf**), первое поле определяет тип записи:

» local – запись определяет политику локального доступа через UNIX-сокеты.
» host – запись определяет политику для сетевого TCP/IP-соединения с использованием SSL и без него. Для обеспечения сетевого доступа необходимо также правильно настроить listen_addresses в **postgresql.conf**.
» hostssl – определяет политику для сетевого соединения с обязательным использованием SSL.
» hostnossl – антипод hostssl.

Второе поле представляет из себя имя базы данных, для которой определяется политика. Имя all означает «все базы», а sameuser – «база, совпадающая с именем пользователя». Можно перечислить несколько имен через запятую. Кроме того, в качестве имени БД можно использовать имя файла со списком баз, разделенным запятыми или пробелами – такое имя должно начинаться с префикса @.

Третье поле – имя пользователя. Здесь тоже поддерживаются списки, а имя all обозначает для всех пользователей. Доступ можно открыть для группы пользователей (role) – для этого перед именем группы следует поставить знак «+».

Следующие ноль (в случае записи local), одно (нотация CIDR[4]) или два поля (адрес и сетевая маска) отведены под сетевой адрес компьютера или подсети, для которых настраивается политика доступа.

Предпоследнее обязательное поле содержит метод авторизации:

» trust – полностью доверяем этому клиенту.
» reject – отказ в доступе.
» ident – доступ по регистрационной записи клиента, часто применяется для локальных соединений. *См. RFC 1413*.
» md5 – авторизация по паролю, зашифрованному с помощью алгоритма md5.

Если клиент использует для доступа библиотеку к *PostgreSQL* старше 7.2, то вместо метода md5 следует использовать метод crypt.

Пароль при желании и значительной степени бесшабашности можно передавать открытым текстом с помощью метода password.

» pam – авторизация с помощью Pluggable Authentication Modules. Этот сервис предоставляется операционной системой.

Подробнее о PAM написано здесь: **http://www.kernel.org/pub/linux/libs/pam/**.

» krb4 и krb5 – авторизация с использованием механизма *Kerberos* версий 4 и 5, соответственно. Это индустриальный стандарт авторизации, и если вы собираетесь его использовать, то, наверное, представляете себе, что это такое.

После имени метода следуют необязательные опции.

postgresql.conf

Настройки в **postgresql.conf** разбиты по группам и подробно задокументированы прямо в файле. Здесь мы коснемся самых важных из них.

Настройка соединений и авторизация (Connections and Authentication)

Политика авторизации настраивается в **pg_hba.conf**. Здесь же собраны в основном технические параметры.

» **Настройка соединений (connection settings)**.

• listen_addresses После настройки **pg_hba.conf** можно и нужно смело устанавливать * – слушаем все интерфейсы, которые есть в наличии. По умолчанию, localhost-запросы принимаются только от локальных пользователей, в том числе и через loopback-интерфейс (127.0.0.1). »

Настройки PostgreSQL для 1С

По адресу **http://v8.1c.ru/overview/postgres_patches_notes.htm** доступны патчи. Это – отличие сервера, поставляемого с «1С:Предприятием 8» от оригинального *PostgreSQL*. Патч **postgresql-1c-8.1.5.patch** несет в себе изменения в исходном файле настройки. Перечислим их:

» Допускаются сетевые соединения:

```
-#listen_addresses = 'localhost'
+listen_addresses = '*'
```

» немного увеличен размер разделяемой памяти: с 8 МБ до 28 МБ:

```
-#shared_buffers = 1000
+shared_buffers = 3500
```

» оценка размера кэша системы изменилась с 8 МБ до 80 МБ:

```
-#effective_cache_size = 1000
+effective_cache_size = 10000
```

» Включен процесс автоматической сборки мусора:

```
-#stats_row_level = off
+stats_row_level = on
-#autovacuum = off
+autovacuum = on
```

» Максимальное среднее число блокировок увеличено более чем в два раза:

```
-#max_locks_per_transaction = 64
+max_locks_per_transaction = 150
```

Мне кажется, что судя по этим изменениям, есть куда оптимизировать и сам продукт, и настройки к *PostgreSQL*.

[3] hba – host-based authentication.
[4] Classless Inter-Domain Routing.

» • port Номер порта, который слушает сервер в ожидании соединений. По умолчанию – 5432.
• max_connections Число клиентов, которые могут подсоединяться к базе данных одновременно, не может быть бесконечным – средний однопроцессорный компьютер может обслуживать не более 100-200 соединений. Данная опция и задает их максимальное число. Если количество клиентов жестко ограничено, то имеет смысл уменьшить этот параметр до минимально возможного значения.
• superuser_reserved_connections Число соединений, которые зарезервированы для суперпользователя, чтобы он всегда мог зайти, разобраться, в чем дело, а затем принять меры. Не стоит совсем отказываться от зарезервированных соединений, и даже одного такого соединения может оказаться недостаточно. Оставьте минимум **2**.

» Безопасность и авторизация (security and authentication).

• authentication_timeout Время ожидания для прохождения авторизации в секундах. По умолчанию – 1 минута. Не позволяет клиенту «зависнуть» и заблокировать ресурс соединения на очень долгое время.
• ssl Разрешает доступ через SSL. Для работы через SSL необходимо создать публичный ключ и сертификат. Это требует некоторых усилий, зато позволяет немного успокоиться по поводу безопасности сетевых соединений.

Управление ресурсами (Resource Consumption)

Правильная оценка имеющихся ресурсов – путь к эффективному планированию. А оно позволяет добиваться поставленной цели, не сильно увеличивая энтропию окружающей среды. Эти очевидные истины реализованы в настройках *PostgreSQL* следующим образом:

» Память (memory).

• shared_buffers Объем совместно используемой памяти, выделяемой *PostgreSQL* для кэширования данных. Определяется числом страниц (shared_buffers) по 8 КБ каждая. Естественно, данные умеет кэшировать не только *PostgreSQL*, но и операционная система, поэтому нет необходимости отводить под кэш всю наличную оперативную память. Оптимальное число shared_buffers зависит от многих факторов, поэтому проще принять следующие ориентиры:

* Обычный настольный ПК с 512 МБ RAM и небольшой базой данных: 8–16 МБ или 1000–2000 страниц.

* Средний сервер 1 Гб RAM и БД около 10 ГБ: 80–160 МБ или 10000–20000 страниц.

* Многопроцессорный сервер с 8 Гб RAM и БД свыше 100 ГБ, обслуживающий несколько сотен активных соединений одновременно – 400 МБ или 50000 страниц.

• work_mem Каждому запросу можно выделить личный участок памяти для сортировки, объединения и других подобных операций. При превышении этого объема сервер начинает использовать временные файлы на диске, что может существенно замедлить скорость обработки запросов. Предел для work_mem можно вычислить, разделив объем доступной памяти (физическая память минус объем, занятый другими программами и буферами shared_buffers) на максимальное число одновременно используемых активных соединений.

При необходимости, например, для выполнения очень объемных операций, допустимый лимит можно изменять прямо во время выполнения запроса, поэтому нет нужды изначально задавать теоретический предел.

• maintenance_work_mem Эта память используется для выполнения операций по сбору статистики (ANALYZE), сборке мусора (VACUUM), созданию индексов (CREATE INDEX) и добавлению внешних ключей. Размер выделяемой под эти операции памяти должен быть сравним с физическим размером самого большого индекса на диске. Как и в случае work_mem, эта переменная может быть установлена прямо во время выполнения запроса.
• max_prepared_transactions Определяет максимальное число подготовленных транзакций (команда PREPARE TRANSACTION). Если эта сущность нигде не используется, то переменную можно занулить.

» Карта неиспользованного пространства (free space map).

Записи при удалении не уничтожаются физически, а только помечаются как удаленные. На этом основана работа сборщика мусора.
• max_fsm_pages Сборщик мусора должен знать, где этот самый мусор находится. Число страниц, отведенных для этих целей должно быть больше, чем число записей, удаленных или измененных между сборками мусора. Если страниц достаточно, выполнение жестких оптимизирующих операций, таких, как VACUUM FULL или REINDEX, никогда и не понадобится. Так как объем требуемой для этого памяти не очень велик (по 6 байт на страницу), то жадничать не стоит. Проще всего узнать необходимое число max_fsm_pages, запустив VACUUM VERBOSE ANALYZE.
• max_fsm_relations Число таблиц, для которых создаются карты неиспользованного пространства. По умолчанию – 1000. В случае большего числа используемых таблиц это значение можно и нужно увеличить, тем более что на каждую таблицу требуется всего по 7 байт.

» Системные ресурсы (kernel resource usage).

• preload_libraries Если для выполнения запроса требуется загрузить какую-либо разделяемую библиотеку, то действует правило: «загружаем при первом использовании», что замедляет исполнение самого первого запроса. Этого можно избежать, загрузив все необходимые библиотеки при старте сервера («память в обмен на скорость»). Хороший кандидат на подгрузку – модуль интерпретатора процедурного языка, используемого в запросах.

» Оценка стоимости сборки мусора (cost-based vacuum delay).

Обычно нет необходимости заглядывать в этот раздел, так как сборка мусора (VACUUM) и анализ (ANALYZE) выполняются достаточно быстро.

» Запись в фоне (background writer).

Начиная с *PostgreSQL 8.0*, вместе с основным сервером также стартует процесс для записи данных в фоне. При выполнении запроса нет необходимости ждать самого акта записи, так как это гарантированно сделает background writer.

Журнал транзакций (Write Ahead Log)

Наличие журнала транзакций или WAL (write ahead log) позволяет увеличить скорость выполнения операций, требующих изменения данных, поскольку информация об изменениях заносится в журнал последовательно, а модификацию таблиц можно отложить до «лучших времен» – своеобразный кэш, только на диске. Если же база данных используется в основном для чтения, то в журнале транзакций особой необходимости нет, но это не повод от него отказываться.

» Настройки (settings).

• fsync По умолчанию эта опция включена (**true**). В этом случае *PostgreSQL* пытается записать данные на диск физически. Это не такая уж и простая операция, так как кэши существуют не только в системе, но и в контроллерах и в дисках. Вполне можно представить себе ситуацию, когда слишком умный диск для увеличения производительности в тестах рапортует о том, что данные записаны, а при перебое с питанием выясняется, что это не так. Сбои самого сервера не приводят к порче данных, но сервер живет в окружении далеко не идеальной операционной системы, которая, в свою очередь, управляет далеко не идеальными физическими устройствами. Вот почему так важно иметь проверенную аппаратуру, источники бесперебойного питания и резервировать данные.

Если вы доверяете своему ПК, то эту опцию можно выключить, сменяв немного безопасности на скорость. Более правильным решением является перенос журнального файла на отдельный быстрый диск.

» Контрольные точки (checkpoints).

По свершению каких-то определенных условий или истечению контрольного времени сервер гарантированно переносит данные, записанные в WAL, непосредственно в таблицы (даже если он очень сильно занят обработкой других запросов).
• checkpoint_segments Объем кэша на диске. Физический объем места на диске, требуемый под кэш, вычисляется по формуле (checkpoint_segments*2+1)*16 МБ. Следует выделить столько, сколько не жалко, осознавая, что 32 сегмента займет на диске свыше 1 ГБ.
• checkpoint_timeout Время, через которое WAL очищается насильно. По умолчанию 300 секунд.
• checkpoint_warning Если кэш на диске заполняется быстрее, чем проходит checkpoint_warning секунд, в журнальном файле появляется предупреждение. Это намек на то, что дисковый кэш следует увеличить.

» Архивация (archiving).

• archive_command Для организации непрерывного резервного копирования (ура настоящим параноикам!) журнал необходимо копировать куда-то еще. Здесь указывается команда, которая будет использоваться системой для копирования данных. Подробности следует искать в документации в разделе Point in Time Recover.

Планирование запросов (Query Planning)

Здесь можно повлиять на логику действия планировщика. Возможно, конкретно для вашей системы значения по умолчанию не оптимальны, но менять их стоит только после серии тестов для выявления более подходящей конфигурации под конкретную платформу и конкретные запросы. В большинстве случаев углубляться в тонкости настроек из этого раздела имеет смысл только в случае очень изощренных запросов.

» Методология планировщика (planner method configuration).

В этом разделе перечислены алгоритмы, которые можно использовать для извлечения данных. В целях тестирования какие-то из них можно отключить.

» Оценочные константы (planner cost constants).

• effective_cache_size В своих планах *PostgreSQL* опирается на кэширование файлов, осуществляемое операционной системой. Этот параметр соответствует максимальному размеру объекта, который может поместиться в системный кэш. Его установка не приводит к увеличению выделяемой памяти. Это значение используется только для оценки.

effective_cache_size можно установить в 1/3 от объема имеющейся в наличии оперативной памяти, если вся она отдана в распоряжение *PostgreSQL*.

Сообщения об ошибках и событиях (Error Reporting and Logging)

Оптимизация возможна только в случае обратной связи. Сервер *PostgreSQL* может многое рассказать о себе. Этот раздел настроек посвящен тому, как правильно его об этом попросить.

» Местоположение журнального файла (where to log).

• log_destination Здесь можно выбрать способ записи в журнальный файл: stderr или syslog. Метод stderr хорош для тестирования, в режиме штатной эксплуатации следует использовать syslog.

При выборе stderr придется также указать директорию для файла журнала (log_directory), его имя (log_filename) и параметры ротации (log_rotation_age, log_rotation_size, log_truncate_on_rotation).

Метод syslog подразумевает настройку через syslog_facility и syslog_iden – идентификационный префикс для сообщений, получаемых от *PostgreSQL*.

» По какому случаю создаем запись (when to log).

Как правило, нет необходимости записывать в дневник информацию о каждом чихе, но при серьезном разбирательстве данные о числе чихов в секунду и их классификация могут подтолкнуть в нужном направлении.

Каждому событию можно присвоить какой-то определенный уровень. Например, уровень PANIC означает, что плохо стало всем, а уровень WARNING сообщает просто о подозрительных, но вполне законных событиях. В порядке возрастания подробности уровни имеют примерно следующую классификацию: PANIC, FATAL, LOG, ERROR, WARNING, NOTICE, INFO, DEBUG[1-5]. Уровень можно установить в процессе выполнения запроса.

Следует осознавать, что журнал необходим при разбирательствах, но его активное использование ведет к уменьшению производительности. Обычно это уменьшение находится в пределах 5% при условии, что журнальный файл и журнал транзакций находятся на разных дисках.

• log_min_messages Характеризует уровень подробности записей в журнальном файле.

• log_error_verbosity Характеризует степень подробности записей. Различаются три степени: TERSE, DEFAULT и VERBOSE.

• client_min_messages Характеризует уровень подробности сообщений, отсылаемых клиенту.

• log_min_error_statement Контролирует уровень подробности записей, создаваемых в результате выполнения SQL-запросов.

» Что именно пишем в журнал (what to log).

В этом разделе настроек перечислены различные возможные источники записей для журнала. Можно записывать информацию о создаваемых соединениях (log_connections), о выполняемых запросах (log_statement), о времени, уходящем на их выполнение (log_duration) и т.п. Параметр log_line_prefix позволяет настроить идентификацию каждой записи по пользователю, IP, базе данных и так далее.

Сбор статистики (Run-Time Statistics)

Есть ложь, гнусная ложь и статистика. База данных врать не научена, поэтому остается только статистика. Этот раздел настроек отвечает за ее сбор. Пока нет необходимости в мониторинге активности базы данных, нет необходимости здесь что-то править.

Для того, чтобы работала автоматическая сборка мусора, должны быть включены опции stats_start_collector и stats_row_level.

Автоматическая сборка мусора (Automatic Vacuuming)

Ну мусор, ну и пусть. Места много – зачем напрягаться, да еще автоматически? В этом есть какая-то логика, но кроме сборки мусора (VACUUM) производится еще и анализ (ANALYZE). Периодическое выполнение команды ANALYZE необходимо для нормального функционирования планировщика. Собранная с помощью этой команды статистика позволяет значительно ускорить выполнение SQL-запросов. Иными словами, сборку мусора делать все равно придется: либо автоматически, либо вручную.

Процесс обычной сборки мусора в *PostgreSQL* (VACUUM без приставки FULL) не блокирует таблицы и может выполняться в фоне, не мешая запросам. Начиная с *PostgreSQL 8.1*, процесс автоматической сборки мусора выделяется в отдельный процесс. Данная группа настроек и контролирует работу этого процесса.

• autovacuum Если вы лучше, чем *PostgreSQL* знаете, когда следует производить сборку мусора, то автоматику можно выключить, хотя лучше ее просто правильно настроить. С другой стороны, сборка мусора оттягивает на себя ресурсы системы, и если это недопустимо, то ее можно отложить на некоторое время.

При настройке службы автоматической сборки мусора и анализа следует понимать, что один из зарезервированных с помощью superuser_reserved_connections слотов может оказаться в нужный момент занят.

• autovacuum_naptime Время в секундах, через которое проверяется необходимость сборки мусора. По умолчанию это происходит раз в минуту.

• autovacuum_vacuum_threshold Порог на число удаленных и измененных записей в любой таблице, по превышению которого происходит сборка мусора (VACUUM). По умолчанию – **1000**, и его вполне можно уменьшить.

• autovacuum_analyze_threshold Порог на число вставленных, удаленных и измененных записей в любой таблице, по превышению которого запускается процесс анализа (ANALYZE). По умолчанию – **500**. Никто не запрещает сделать его поменьше.

• autovacuum_vacuum_scale_factor Процент измененных и удаленных записей по отношению к общему числу строк в таблице, по превышению которого запускается сборка мусора. По умолчанию – **0.4**.

• autovacuum_analyze_scale_factor То же, что и предыдущая переменная, но по отношению к анализу. Значение по умолчанию равно **0.2**.

Настройки для пользователя по умолчанию (Client Connection Defaults)

Настройки из этой группы задают некоторые значения по умолчанию для подсоединившегося к базе данных клиента и не влияют на производительность самого сервера. Исключением является разве что переменная statement_timeout, которая выставляет ограничение времени на выполнение запроса (в миллисекундах), да и то – не активированная »

PostgreSQL в лицах: Алексей Харитонов

Визитка LXF: Руководитель отдела продвижения экономических программ фирмы «1С»

Некоторое время назад фирма «1С» объявила о выходе версии 8.1 платформы «1С:Предприятие», серверная часть которой теперь работает под управлением Linux, а в качестве хранилища данных может использоваться и *PostgreSQL* (модифицированная версия).

Можно долго и эмоционально говорить о продуктах фирмы «1С», но они реальны, и реальность эта повседневна. А то, что эта реальность стала использовать продукты с открытым кодом – это явный плюс. Для выяснения подробностей мы связались с представителем фирмы **Алексеем Харитоновым** через стандартный e-mail: **1c@1c.ru**.

На прилагаемом диске вы можете найти полный текст интервью, прокомментированный Федором Сигаевым (LXF86) и Олегом Бартуновым (LXF85) – отечественными разработчиками, внесшими значительный вклад в развитие *PostgreSQL*.

Евгений Балдин (ЕМБ): Почему вы выбрали именно *PostgreSQL*? Конечно, лет пять назад особых альтернатив среди свободных продуктов не было. Но сейчас подросла в своих возможностях *MySQL*, *Firebird* восстал из пепла Interbase, да и смысла зацикливаться на открытых реализациях для 1С особого нет. Или есть?

Алексей Харитонов (АлХ): Нам требовалась DBMS, работающая под ОС Linux, лицензия на которую предусматривала бы возможность ее свободного (без отчислений) распространения и самостоятельной модификации. Такой вариант мы считаем полезным для определенного сегмента рынка. Из DBMS для Linux, лицензируемых на таких условиях, мы оценили *PostgreSQL* как наиболее качественную и эффективную.

ЕМБ: Какие изменения были внесены в базовый код *PostgreSQL*? Будут ли эти изменения интегрированы в основное дерево проекта *PostgreSQL*?

АлХ: Основная причина, по которой появилась необходимость модифицировать *PostgreSQL 8.1* – это то, что оригинальная версия не поддерживает сравнение строк, нечувствительное к регистру символов. Внесенные в *PostgreSQL* изменения позволили эффективно (быстро) работать с русскими символами в кодировке Unicode. Кроме того, конфигурационные файлы, которые получаются после установки *PostgreSQL 8.1*, модифицированного для работы с «1С:Предприятием», несколько отличаются от оригинальных. Это также сделано для лучшего согласования *PostgreSQL 8.1* с «1С:Предприятием» 8.1.

Мы взаимодействуем с разработчиками *PostgreSQL*. Дополнительные модули расширения и необходимые изменения к БД были подготовлены членами команды разработчиков *PostgreSQL* Олегом Бартуновым и Федором Сигаевым. Включение этих изменений в основную ветку будет еще обсуждаться.

ЕМБ: Что с точки зрения «1С» нужно добавить в *PostgreSQL*, чтобы улучшить его функциональность?

АлХ: Критически важные для нас исправления мы внесли в патч, часть этих исправлений уже реализована в *PostgreSQL* 8.2. Прежде всего мы рекомендовали бы внести в основной проект все сделанные нами изменения, поскольку они необходимы для работы различных систем (не только «1С:Предприятия») с русским языком. Есть и другие пожелания – пока мы обсуждаем их с разработчиками *PostgreSQL*.

ЕМБ: Есть ли в планах «1С» сотрудничество с сообществом Open Source, ну, или, хотя бы, использование других открытых продуктов для целей бизнеса?

АлХ: Примеры использования нами открытых продуктов не ограничиваются *PostgreSQL* – так, в «1С:Предприятии 8.1» расширенная поддержка XML реализована на основе библиотек Apache Xerces и Apache Xalan. Для интернационализации используется библиотека ICU.

В качестве одного из возможных направлений бизнес-сотрудничества мы сейчас рассматриваем варианты лицензирования Linux-версии сервера приложений «1С:Предприятия 8.1» для OEM-поставки с дистрибутивами Linux.

по умолчанию. Но если хочется настроить локаль (хотя это личное дело клиента) или порядок выбора объектов, относящихся к различным пространствам имен, то здесь можно что-то поправить.

Управление блокировками (Lock Management)

Очевидно, что блокировок следует избегать всячески, причем делать это надо начинать на этапе проектирования базы данных. К сожалению, реальная жизнь отличается от планов.

- deadlock_timeout Взаимные блокировки (deadlock) – это чрезвычайно уродливое явление, при котором вошедшие в клинч процессы ожидают освобождения ресурсов, которые сами же и захватили. *PostgreSQL* умеет разрешать эту проблему путем насильного прерывания одного из процессов. Проверка на deadlock – это довольно длительная процедура, поэтому, прежде чем начать такую проверку, сервер выжидает указанное время. По умолчанию – 1000 миллисекунд. Для загруженных серверов имеет смысл увеличить это значение.
- max_locks_per_transaction Вопреки своему названию, это не жесткий лимит на число блокировок, осуществляемых в пределах транзакций. Этот параметр входит в формулу, устанавливающую предел на число одновременно существующих блокировок, то есть это скорее максимальное среднее:

```
max_locks_per_transaction * (max_connections + max_prepared_
transactions)
```

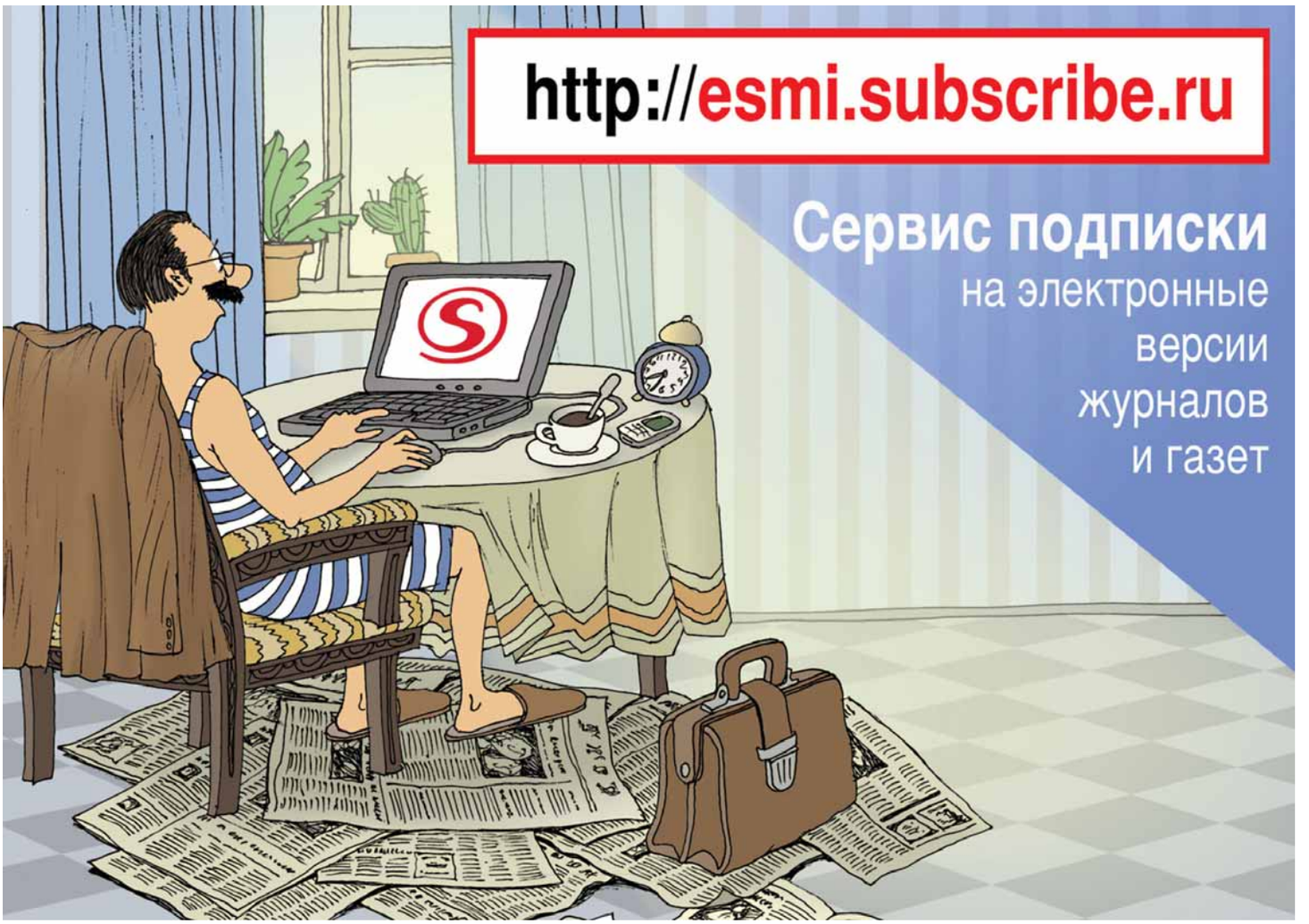
В документации сказано, что **64** (число стоящее по умолчанию) – это исторически проверенный предел, и чтобы превзойти его, требуются определенные усилия.

О том, что думать тоже надо

Можно идеально настроить сервер, регулярно проводить сборку мусора, можно закупить самое дорогое оборудование и поставить рядом с ним дизельный генератор. Но если таблицы и отношения между ними создавались без плана, а запросы задаются криво, то проблемы гарантированно будут.

Выполнение запросов следует проверять с помощью команды EXPLAIN ANALYZE, которая по полочкам разложит, как ищется, сортируется, объединяется и группируется информация, какие для этого использовались алгоритмы, и какие индексы были задействованы. Для любителей картинок pgAdmin III имеет графический интерфейс к этой команде. Ни в коем случае нельзя пренебрегать индексами и по возможности следует избегать блокировок. LXF

» **Через месяц** Полнотекстовый поиск и другие крупные дополнительные модули *PostgreSQL*.



В «Системном администраторе» вы не прочтете о:

– котировках валют
– сплетнях
– погоде
– политике
– развлечениях

В вашем распоряжении:

– опыт лучших IT-специалистов
– новые идеи и полезные советы
– самые эффективные решения в области системного и сетевого администрирования

Подпишитесь сейчас!

Роспечать – 20780, 81655
Пресса России – 87836
Online-подписка – www.linuxcenter.ru

Время подписки ограничено!

Путеводитель по

ЧАСТЬ 7 Подобно Золушке, ставшей принцессой, документ *LaTeX* может несколько раз сменить свой класс за время существования. Многообразие классов *LaTeX* способно смутить любого – но **Евгений Балдин** предлагает их полную классификацию.

Классы у людей определяются выбором оружия…
Википедия. Из описания игры Tremulous.

Класс документа – это первое, что требуется указать при наборе. В то же время первое вовсе не значит важное. С заключительным выбором класса почти всегда можно повременить до окончания основного набора. С другой стороны, выбирать всё равно когда-нибудь придётся, так что нелишне знать, что может нам предложить *LaTeX*.

Зачем нужны эти классы?

Класс документа выбирается с помощью команды documentclass. Её нужно и можно выполнить ровно один раз в самом начале документа:

```
\documentclass[a4paper,12pt]{article}
```

В качестве обязательного аргумента указывается имя класса, которому через запятую передаются необязательные параметры. Класс определяется в файле с расширением **.cls**. В дистрибутиве *LaTeX TeX Live 2005* присутствует 175 уникальных файлов с подобным расширением. Естественно, это далеко не все существующие на белом свете классы, тем более, что никто не мешает вам создать свой личный класс. Но для начала лучше все же воспользоваться одним из имеющихся.

Класс определяет вид и структуру документа. Класс – это база, которую можно править с помощью подключаемых стилевых файлов. В классе задаётся геометрия страницы и определяются команды секционирования. Сам по себе класс может быть как просто небольшой модификацией уже имеющегося, так и принципиально новой реализацией представления печатного или электронного ТеХнического слова. В качестве примера последнего можно привести молодой (2007 год) пока пакет papertex, который позволяет применять *LaTeX* в деле вёрстки газет[1], то есть для того, для чего *LaTeX* в проекте вовсе не предназначался. Сам пакет можно найти на любом CTAN-архиве в директории **{CTAN}/macros/LaTeX/contrib/papertex/**.

my papertex edition http://nachollopis.com

paperTeX

xi Februarii MMVII ‖ 10:19h

First heading. Aenean sem purus, blandit bibendum, gravida sed.

12:34 h — First text. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In vitae augue. Morbi fermentum, felis accumsan rhoncus malesuada, metus eros adipiscing dui, vel eleifend nibh velit ut felis. Pellentesque elementum massa at nisi dapibus mollis. Etiam cursus. Mauris gravida nibh ut sapien. Vestibulum pretium, felis et.

This is the caption of the main image on the front page.

SECTION A

Second heading. Fusce porttitor erat aliquam felis. Integer scelerisque enim at augue.

Second subheading. Facilisis sit amet, elementum nec, interdum vitae, nunc.

10:23 h — Second text. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In vitae augue. Morbi fermentum, felis accumsan rhoncus malesuada, metus eros adipiscing dui, vel eleifend nibh velit ut felis. Pellentesque elementum massa at nisi dapibus mollis. Etiam cursus. Mauris gravida nibh ut sapien. Vestibulum pretium, felis et.

SECTION B

Third heading. Donec lectus enim, dapibus sit amet, condimentum non, viverra eu, augue.

Third subheading. Nunc dictum fermentum nunc. Suspendisse est. Nulla bibendum lacus vel tortor.

10:02 h — Third text. Curabitur imperdiet, est vel commodo eleifend, mi urna vulputate lorem, at rhoncus diam augue vitae magna. In nisi. Vivamus elementum blandit sem. Cras sit amet justo. Curabitur diam arcu, elementum commodo, tincidunt vel, dapibus id, lectus. In egestas nibh nec leo. Pellentesque ut nisi sed purus rhoncus vehicula. Donec commodo magna.

MAIN INDEX

Index item 1. Cras molestie dui sed lectus. In purus justo, lacinia sit amet. p. ??

Index item 2. Cras feugiat, arcu a tincidunt ornare, sem augue mattis erat. p. ??

Index item 3. Sed sit amet velit. Duis sit amet nunc. Aenean vehicula commodo est. Aliquam dignissim tempor enim. p. ??

Index item 4. Curabitur imperdiet, est vel commodo eleifend, mi urna vulputate lorem. p. ??

paperTeX
EDITORS
Ignacio Llopis
lloptor@gmail.com
http://nachollopis.com

WEATHER FORECAST

TODAY 13 ‖ 9 °C | TOMORROW 15 ‖ 11 °C | FRIDAY 12 ‖ 6 °C

› (Рис. 1) Класс papertex — пример от автора класса Ignacio Llopis.

Классовая база

Исторически сложилось так, что *LaTeX* начался с 6 классов: article (статья), book (книга), report (отчёт), proc (доклад), letter (письмо) и slides (слайды).

По идее, статьи следовало набирать в article. В этом классе определены команды секционирования вплоть до section (раздел). Одним из желательных элементов оформления является предисловие (окружение abstract). В классе book присутствует расширенный набор команд секционирования, в который добавлена команда chapter (глава). Также в классе book присутствует базовый набор команд для оформления титульного листа, предисловия и оглавления. Класс для создания отчёта report является упрощённой версией класса book. Отчёты – те же книги, только читают их по необходимости, а не по желанию. Класс proc, предназначенный для создания тезисов докладов, в свою очередь является модификацией класса article, причём основное отличие состоит в обязательной двухколоночной печати. Для написания писем был создан класс letter. Набор команд в этом классе существенно отличается от уже перечисленных, в частности, для писем нет нужды в командах секционирования. Класс slides – простой и быстрый способ сделать презентацию. С помощью этого класса не удастся создать пёстрого фона и головокружительных эффектов смены слайдов, зато он позволяет сосредоточиться на основном – на тексте.

Особняком от этих классов стоит класс minimal, который является «болванкой» для создания и тестирования новых классов и идей. В классе minimal не определено никаких специальных команд – минимум есть минимум. В качестве побочного эффекта документ, выбирающий этот класс, транслируется *LaTeX* значительно быстрее.

С этих классов всё начиналось, но на них ничего не закончилось. Следует понимать, что базовые классы далеко не так хороши, как хотелось бы. После трансляции текста сразу возникает желание взять в руки «электронный надфиль» и пройтись по настройкам класса. Это цена за то, что перечисленные классы являются базовыми. На них ссылается и их модифицирует множество других классов и пакетов, поэтому их развитие было заморожено. Для начального набора сгодится и

[1] На текущий момент этот класс не годится для вёрстки чего-нибудь более серьёзного, чем школьная газета, но ведь надо начинать с простого.

Месяц назад Мы овладели искусством вёрстки в *LaTeX* – или,

классам LaTeX

это, но для конечной вёрстки лучше подобрать что-то более подходящее, или придётся серьёзно модифицировать значения по умолчанию.

Часто набор необязательных параметров для стандартных классах используется и в других классах, например, в целях совместимости. Некоторые из полезных опций перечислены ниже:

- **10pt|11pt|12pt** – установка базового размера шрифта. Обычно этих трёх значений хватает.
- **a4paper** – установка размера листа бумаги. Следует использовать всегда, так как по умолчанию *LaTeX* использует размер листа letter.
- **draft** – режим черновой печати для «отлавливания» проблем вёрстки. В этом режиме не внедряются картинки (вместо них вставляются прямоугольники нужного размера) и отмечаются строчки, где алгоритм разбиения абзаца на строки даёт осечку.
- **oneside|twoside** – форматирование документа для односторонней и двухсторонней печати, соответственно.
- **twocolumn** – печать в две колонки.

Классификация

Сразу предупрежу: число классов постоянно растёт, поэтому не следует думать, что все имеющиеся варианты исчерпываются классами, перечисленными ниже.

Модификации и улучшения базы

Стандартные классы не нравятся всем, и всякий старается их улучшить. Кто-то убирает какой-то конкретный недостаток, как это сделано в наборе классов extsizes (extarticle, extbook, extletter, extproc, extreport), которые отличаются от стандартных только возможностью указать базовый размер шрифта, отличный от обычного 10-12pt[3]. Есть наборы классов, которые делались с какой-то определённой целью. Примером такого подхода являются классы от AMS (amsart, amsbook, amsproc), которые используются для материалов, предназначенных для публикации в журналах Американского математического сообщества. Классы из набора ntgclass[3] представляют из себя «героическую» попытку немецкоговорящих голландцев сделать то же, что и в стандарте, но существенно разными способами.

KOMA-Script

В последнее время всё больше внимания обращает на себя набор классов KOMA-script. В этот раз хорошо постарались немцы. Следует учитывать, что европейские традиции полиграфии (в основном французские), всё-таки ближе к нам, чем американские, на которые традиционно ориентировалось *LaTeX*-сообщество. Для статей предполагается использовать scrartcl, для книг scrbook, для писем scrlttr2, а для отчётов scrreprt.

В отличие от стандарта, классы из KOMA-script позволяют использовать базовые размеры шрифта в 9pt, 14pt и 17pt. Огромные поля, имеющие место в стандартных классах, в классах KOMA-script значительно уменьшены. Претерпели изменения и другие элементы. Если оформление по умолчанию не кажется адекватным, то KOMA-script предоставляет обширный набор высокоуровневых настроек. Подробная документация на более чем двухстах страницах «The KOMA-Script bundle» (**scrguien.pdf**) позволяет подстроить все необходимые параметры.

NCC

Ещё один вариант замены стандартных классов – это использование пакета NCC. Очень подробно об этом пакете написано в замечательной книге от создателя NCC Александра И. Роженко: «*Искусство верстки*[4] *в LaTeX'e*», 2005 (ISBN 5-901548-25-6).

Для использования следует загрузить класс ncc и передать ему желаемый стиль оформления в качестве параметра: article (статья – используется по умолчанию), preprint (препринт), book (монография) или report (отчёт). Дальнейшие подробности об использовании этого класса можно почерпнуть в краткой инструкции к пакету: **nccLaTeX.pdf**.

Поддерживаем стандарты

Стандарт подразумевает наличие подробного описания, которое и является его сущностью. То, что написано на бумаге в виде набора непротиворечивых правил, может быть переведено на язык машины. Далее можно забыть про эти правила, так как помнить все нюансы – работа для машины[5] (**Рис. 2**).

Константин Кориков создал и активно поддерживает пакет *eskdx*, который представляет из себя набор классов и стилей, предназначенный для вёрстки документации в соответствии с требованиями «Единой системы конструкторской документации». Основу коллекции составляют три класса: eskdtext (для текстовой документации), eskdbtab (для чертежей и схем) и eskdgraph (для документов, разбитых на графы). Внятная документация на русском (**eskdx.pdf**) приятно дополняет картину. Пакет можно найти на CTAN в директории **{CTAN}/macros/LaTeX/contrib/eskdx/**, а его домашняя страничка расположена по адресу: **http://lostclus.linux.kiev.ua/eskdx/**.

eskdx – относительно молодой пакет. Ранее аналогичная попытка была предпринята Вячеславом Фёдоровым, в результате которой на свет появился пакет *eskd* (без «x» на конце). В отличии от класса Константина Корикова, класс eskd.cls требует обязательной установки шрифтов из коллекции pscyr[6].

Стандарт для написания документов, описывающих стандарты. Что может быть ещё более стандартным? Класс isov2.cls из пакета *iso* является стандартом для стандартов. Документация к пакету **isoman.pdf** подробно описывает все технические тонкости в деле подготовки документов по стандартам ISO. Аналогично, для создания документации ISO 10303 есть свой пакет iso10303.

Пишем письма

Класс letter является стандартным для *LaTeX* и, как следствие, никто им не пользуется. Часто стандарт для написания писем создаётся автором самостоятельно. Так, например, любит делать Кнут. Это очень неплохо работает в силу того, что структура письма не слишком сложна. Как следствие, в *LaTeX* имеется огромное число альтернатив для letter.

С точки зрения английского FAQ по *LaTeX* (**http://www.tex.ac.uk/faq**) класс newlfm является наиболее продвинутым. lfm расшифровывается

[2] Кроме 10pt, 11pt и 12pt классы из набора extsizes поддерживают 8pt, 9pt, 14pt, 17pt и 20pt. Смена размера базового шрифта приводит к принципиально иному дизайну документа

[3] В набор ntgclass входят классы для набора статей (artikel1, artikel2 и artikel3), для набора книг (boek и boek3), писем (brief) и отчётов (rapport1 и rapport3).

[4] Да, да, именно «верстки» – букву «ё» опять обидели.

[5] То есть тупая, нудная и неинтересная.

[6] Шрифты из коллекции pscyr авторами больше не поддерживаются и не развиваются. Основные проблемы этого пакета не технические, а лицензионные, в связи с чем этот пакет отсутствует в дистрибутивах LaTeX. Впрочем, в дополнение к абсолютно не решаемым лицензионным, в них хватает и технических проблем. Последнюю версию пакета можно найти по адресу: **ftp://ftp.vsu.ru/pub/tex/font-packs/pscyr/**.

СОДЕРЖАНИЕ

1 Общие сведения 4
1.1 О коллекции **eskdx** 4
1.2 Возможности коллекции 4

2 Базовые принципы использования 5
2.1 Пример простого документа 5
2.2 Опции классов 6
2.2.1 Общие опции всех классов 6
2.2.2 Опции класса **eskdtext** 8
2.2.3 Опции класса **eskdgraph** 9
2.2.4 Опции класса **eskdtab** 10
2.3 Информация о документе 11
2.4 Титульный лист 13
2.5 Заполнение граф основной надписи и дополнительных граф 14
2.6 Рубрикация 17
2.7 Пояснения символов, входящих в формулу 18
2.8 Лист регистрации изменений 18
2.9 Чертежи и схемы 18
2.10 Спецификация 19
2.11 Спецификация при плазовом методе 19
2.12 Лист утверждения 19
2.13 Количество рисунков, таблиц, приложений, и т.д. 20

3 Тонкая настройка 21
3.1 Управление стилями страниц 21
3.2 Настройка шрифтов 22
3.3 Настройка титульного листа 23
3.4 Управление заголовками рубрикации 24

Изм.	Лист	№ докум.	Подп.	Дата				
Разраб.		Коряков К. В.			*Коллекция eskdx v0.97 Руководство пользователя*	Лит.	Лист 2	Листов 32
Пров.								
Н. контр.								
Утв.								

(Рис. 2) Страница документации к eskdx.

как letter (письмо), fax (факс) и memorandum (служебная записка). Документация представляет из себя текстовый README и набор примеров использования.

Хорошо документированный класс akletter также является хорошим шаблоном для старта. Документация **lettereng.pdf**, кроме краткой инструкции, также включает и формальное описание структуры письма.

Упомянутый выше пакет KOMA-script предоставляет прекрасную замену стандартному классу letter в виде scrlttr2. Набор классов ntgclass так же предоставляет свой вариант в виде класса brief.

Кроме более-менее общих решений, полно и частных. Например, для внутренней переписки университета города Падуя есть специальный пакет cdpbundl, содержащий целых три класса.

Верстаем книги

Написание книги – это очень долгий процесс, и первоначальную «набивку» текста можно начать со стандартного класса book. С другой стороны, структура книги может быть очень сложной, и правильный выбор базового класса позволит несколько облегчить процесс созидания.

В качестве улучшенного стандартного класса book можно использовать класс octavo. Класс scrbook из KOMA-script также является хорошей альтернативой для book. Структура и основные команды копируют стандартный класс. Значения же параметров по умолчанию более адекватны для европейской полиграфии.

Описание класса memoir (**memman.pdf**) представляет из себя книгу о создании книги, превышающую по объёму триста страниц. Там есть всё, начиная от формальной структуры печатной книги, советов по оформлению электронных копий, и заканчивая собственно описанием класса. Все элементы структуры и управляющие размеры показаны в виде рисунков и схем. Класс не является надстройкой над чем-либо – это произведение искусства, созданное с нуля. Документацию следует пролистать хотя бы просто для ознакомления. Класс развивается до сих пор. Новые возможности описываются в дополнении к основной документации (**memmanadd.pdf**), и на текущий момент дополнение почти достигло объёма в сто страниц.

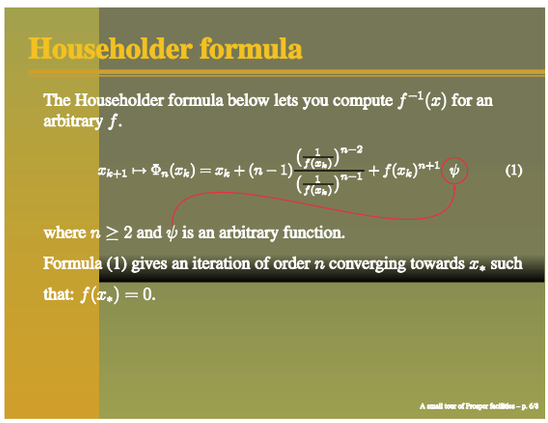
Если не требуется написать книгу, а нужно распечатать мегабайтный текст на дешёвеньком принтере в режиме экономии тонера, то для этого дела вполне может подойти класс sffms – простенько и строки через два интервала.

Создаём отчёты

Отчёт – не книга, но и здесь есть свои правила и структура. Для начала можно воспользоваться scrreprt из KOMA-script, как замена стандартному классу report.

Инженер-электронщик Эли Биллауэр [Eli Billauer] сделал *LaTeX*-класс для бумаг в HiTech-стиле и назвал его, соответственно, hitec. Простенько и со вкусом. Есть, естественно, и частные решения. Хочется заключить контракт с американским правительством – стандартная форма 298, обеспечиваемая классом sffms, будет вполне кстати. Класс manual из пакета nassflow даст возможность пообщаться со структурой под названием «Center for the Automation of Weapon and Command Systems, Royal Netherlands Navy» [Центр автоматизации вооружения и командных систем Королевских ВМФ, Нидерланды].

Делаем презентации



(Рис. 3) *prosper* в действии.

В начале предполагалось, что цель презентации состоит в распространении нужной информации от одного человека ко многим. Поэтому во главу угла ставился текст, а «украшательства» сводились к простой рамке. Стандартные классы seminar и sides вполне годились для этого.

Но время суровых докладчиков прошло, и «рюшечки» вышли на первый план. *LaTeX* может предоставить и «рюшечки», но лучше всё-таки помнить о смысле.

Класс prosper создан как улучшенный seminar и поддерживает не только оверлеи, гиперссылки и шаблоны оформления, но и «стандартный» набор динамических эффектов доступных через формат pdf[7]. Вспомогательный класс ppr-prv позволяет создать печатную версию электронных слайдов класса prosper.

В пакет texpower входит класс powersem, который, по сути дела, просто загружает slides, а всю работу по созданию презентации выполняет стиль texpower. Возможности этого пакета сравнимы с prosper – просто он немного другой.

В последнее время в деле создания презентации всё популярнее и популярнее становится относительно молодой, но довольно мощный класс beamer. Активная поддержка сообщества при создании этого пакета позволила автору beamer Тилу Тантау (Till Tantau) собрать в одном месте немало тем для слайдов. Ключевой особенностью этих тем является разнообразие. Наличие широкого выбора стандартных тем позволяет быстро выбрать обрамление для презентации. Более чем 200 страниц документации ускоряют решение любой возникшей проблемы.

Для создания презентаций можно настроить даже WYSIWYM-редактор *LyX*. Для этого следует воспользоваться ещё одним достаточно молодым, но уже вполне функциональным классом powerdot, в комплекте с которым идут настройки для *LyX*.

Класс talk, в отличие от упомянутых выше пакетов, позволяет пользователю определить более одного стиля слайдов для презентации. Резкая смена стиля во время доклада – иногда бывает нужно и такое.

Для создания настенного постера в первом приближении мож-

[7] Для просмотра динамических эффектов требуется Acrobat Reader, раскрытый на полный экран.

Условия отбора $\psi' \to J/\psi\pi^+\pi^-$

- $\cos(\alpha)<0.75$, где α – угол между мезонами и прямой, вдоль которой двигались электроны
- импульс $p_{\pi^+;\pi^-}<0.5$ ГэВ
- $3.0\,\text{ГэВ} < M_{rec} < 3.2\,\text{ГэВ}$, где M_{rec} – масса отдачи для $\pi^+\pi^-$
- косинус угла между мезонами $\cos(\pi^+;\pi^-) < 0.9$

5/7

› (Рис. 4) *beamer* – справится даже школьник. Изготовлено учеником 11 класса А.Ф. Зайковым самостоятельно. На ошибки не фыркаем! Школьник всё-таки.

но воспользоваться пакетом a0poster, который позволяет работать с большими форматами бумаги. Канонического класса, который бы решал все проблемы при изготовлении постера, в *LaTeX* на текущий момент не существует. Возможно, ближе всего к идеалу подошёл класс sciposter из одноимённого пакета.

Защищаем диссертации

Раньше всюду были курсовые и дипломные работы, а теперь, куда ни плюнь, везде диссертации. Каждый уважающий, не сильно уважающий и вообще не уважающий себя университет имеет свой уникальный стиль оформления диссертации. Если хочется написать свой класс, то в качестве отправной точки можно выбрать ucthesis от Калифорнийского университета (UC Berkeley).

Станислав Кручинин озадачился судьбами русскоговорящих диссертантов и создал класс disser. Пакет с одноимённым названием можно взять на CTAN в директории **{CTAN}/macros/LaTeX/contrib/disser**. Следует учесть, что пользовательская документация на текущий момент отсутствует. С другой стороны, диссертанты, по идее, люди неглупые и разберутся в имеющихся примерах.

Организуем резюме

Написание резюме или curriculum vitae – довольно популярный вид деятельности в современном мире. Для резюме нет общепринятого стандарта, но некоторые указания существуют.

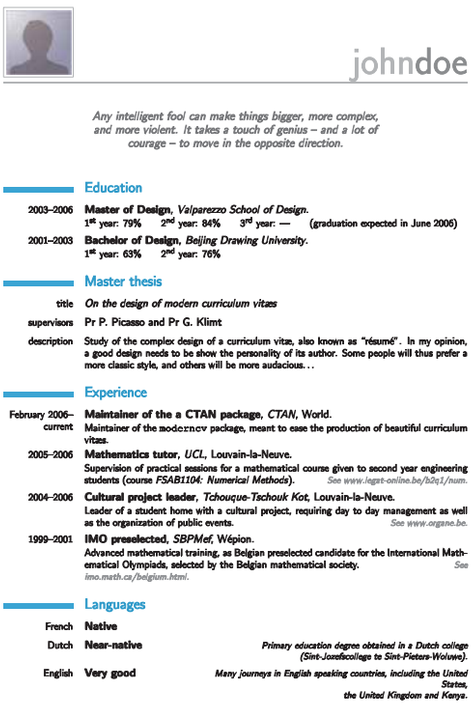
Европейская комиссия рекомендует определённый формат для составления резюме, и этот формат полностью реализуется с помощью класса europecv. Класс vita представляет собой конструктор для создания резюме. Несмотря на отсутствие документации, разобраться с ним по имеющимся примерам для IT-специалиста и певца не составит сложности. Класс curve, напротив, обладает качественной документацией. Механизм рубрик позволяет классу curve поддерживать несколько резюме разной направленности и легко переключаться между ними. Современный класс morderncv рекомендуется как гибкое и простое средство создания резюме как современного вида, так и классической формы. Пакет можно взять на CTAN в директории **{CTAN}/macros/LaTeX/contrib/moderncv**.

Следует отметить, что классы определяют многое, но далеко не всё. Поэтому после выбора класса можно подключить стили, которые серьёзно поменяют внешний вид документа. Стиль currvita позволяет создавать резюме в окружении стандартных классов.

Журнальные и конференционные классы

Каждый серьёзный научный журнал и крупная конференция имеет свой *LaTeX*-класс. Обычно этот класс лежит где-то на официальном сайте, например, журналы, издаваемые издательством МАИК «*Наука/Интерпериодика*», должны следовать правилам, выложенным здесь: http://www.maik.ru/pub/tex/. Но довольно много журнальных классов можно найти в стандартном дистрибутиве *LaTeX*. Например, класс asaetr используется в American Society for Agricultural Engineers (ASAE). Maple Technical Newsletter можно создавать с помощью класса mtn. Классом jpsj2 отметились японцы. Из русскоязычных журналов замечен только «Сибирский журнал вычислительной математики» – класс sibjnm.

Следует отметить класс elsart, обязательный для подготовки журнальных публикаций в издательстве Elsevier. Класс nature позволит подготовить pdf-файл для журнала Nature. Часто при создании публикаций для журналов или конференций используются небольшие модификации класса revtex4.

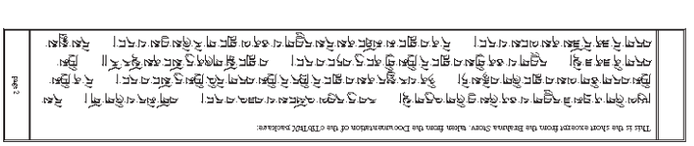


› (Рис. 5) Класс *moderncv*.

Для объединения разных документов в один, например, для оформления трудов конференции, может пригодиться класс combine.

Всякая всячина

Далеко не все классы подчиняются уже перечисленной классификации. За рамки темы вышли классы для составления календарей, обложек для CD (cd), вопросников (qcm), объявлений о занятиях (assignment), концертных программок (ConcProg), программ курсов (courseoutline и coursepaper), рабочего журнала для биологов (labbook), пьес (stage), тибетских карточек (pecha), карточек для запоминания иностранных слов (flashcards), и для многого другого. Старые и давно не поддерживаемые классы могут не собраться в новом окружении, но исходные тексты доступны, так что их всегда можно довести до необходимой кондиции.



› (Рис. 6) Что-то определённо тибетское – класс *pecha*.

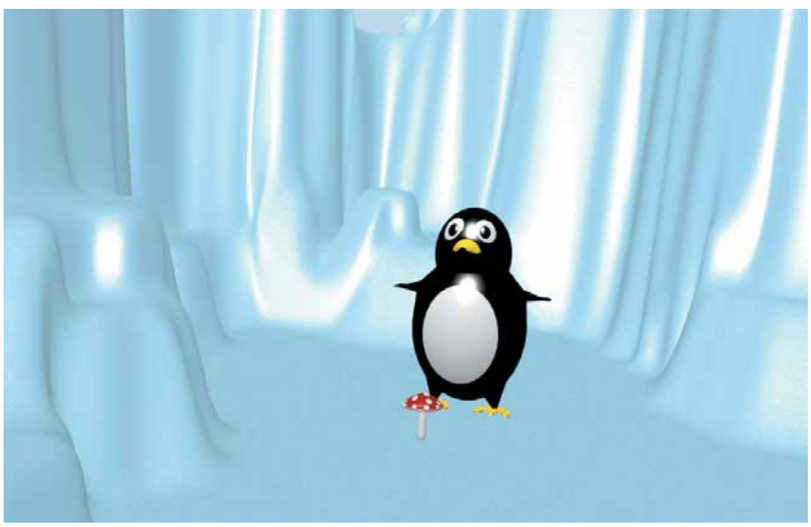
Заключение

Классов существует много – места в статье мало. Малая толика из упомянутых классов будет рассмотрена в последующих статьях цикла. Это не является проблемой, так как в чём-чём, а в отсутствии документации к пакетам *LaTeX* обвинить невозможно – ну, почти. LXF

» Через месяц Мы познакомимся с некоторыми из классов поближе – и научимся создавать красочные презентации.

Анимация и

ЧАСТЬ 3 Сегодня **Андрей Прахов** вдохнет жизнь в созданный нами мирок, раскрасив его обитателей и научив их двигаться в соответствии с законами физики.



На прошлых уроках мы с вами познакомились с основами моделирования в *Blender* и даже сваяли симпатичную фигурку пингвина. Но, как вы понимаете, возможности программы не ограничиваются банальным моделингом. В этот раз мы познакомимся с различными с видами анимации, продолжим разговор о текстурировании, попробуем на ощупь физику. Вдохнем же жизнь в нашу картинку!

Прежде чем мы приступим к анимации, необходимо разобраться с возможностями программы в этой области.

Задача №1.

Предположим, имеются пункты «А» и «В», необходимо переместить объект из одной точки в другую.

Подобный метод анимации в трехмерной графике получил название «ключевой». Смысл его в том, что указываются начальное и конечное положение объекта, а программа достраивает промежуточные значения. Это самый простой и распространенный вид анимации. С его помощью можно не только перемещать и вращать объекты, но и масштабировать их, работать с материалами и изменять внутреннюю структуру. Для работы с данным инструментом существует окно Timeline (**рис. 1**)

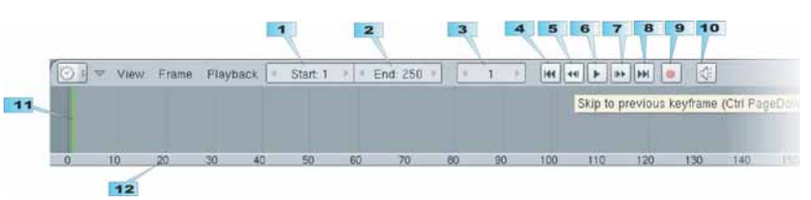


Рисунок 1.

1. Начальное значение кадра;
2. Конечное значение кадра;
3. Пошаговая перемотка. Для быстрого перехода в нужное место введите сюда свое значение;
4. Переход в начало линейки;
5. Переход на ключ назад;
6. Проигрывание анимации. Приятная особенность *Blender*'а в том, что он позволяет редактировать сцену без остановки проигрывания;
7. Переход на ключ вперед;
8. Переход в конец анимации;
9. «Автозапись». Автоматическое создание ключей в зависимости от манипуляций над объектами;
10. Отключение/включение звука при проигрывании анимации;
11. Курсор текущего кадра;
12. Градация линейки.

Принцип работы с ключами очень простой и я продемонстрирую его на следующем примере. Создайте куб и откройте окно Timeline в удобном для вас месте. Находясь в режиме Object Mode и при выделенной фигуре, нажмите I для вызова меню вставки анимационных ключей. Выберите верхнее значение Loc. Сдвиньте анимационный курсор (№11, **рис. 1**) на любое количество кадров вперед и активируйте клавишу записи (№9, **рис. 1**). Передвиньте куб в произвольное место. На линейке появятся желтые полоски, обозначающие установленные ключевые кадры. Если теперь проиграть анимацию, вы увидите движение объекта из одной точки в другую.

Задача №2.

Передвинуть объект из точки «А» в точку «В» вдоль некоторой кривой.

Первый метод обеспечивает линейную анимацию, что и было продемонстрировано в примере выше. Более точную настройку анимации в пространстве и во времени предлагает специальный инструмент *Blender'a* – IPO Curve Editor.

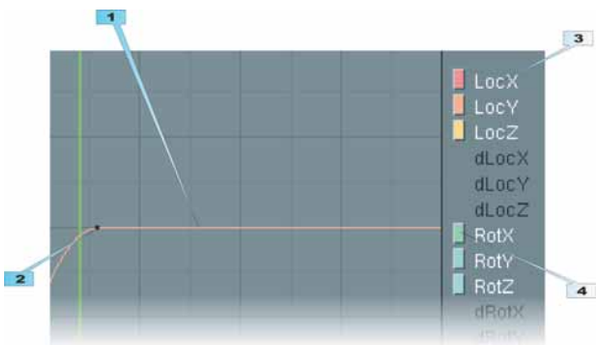
Позволю себе немного отвлечься от темы и напомнить, что программа имеет ряд предустановленных типов расположения окон – их можно выбрать из выпадающего списка в верхней части экрана. Быстрые клавиши перехода от темы к теме – Ctrl+курсор влево или вправо.

Перейдите в режим отображения окон SR:1 - Animation. Эта тема, как нетрудно догадаться, наиболее удобна для создания анимации. В

Месяц назад Мы занимались моделингом и создали выводок пингвинов.

текстурирование

правом углу расположено окно IPO Curve Editor (**рис. 2**):



1. Область, отображающая кривые для манипуляции конкретными переменными;
2. Точка, являющаяся ключевым кадром в анимации;
3. Название свойства трансформации объекта;
4. Кнопка активации конкретного показателя.

Работа с этим редактором требует определенного пространственного мышления, так как кривые представляют управление объектом в схематическом виде. Тем не менее, в работе это окно практически ничем не отличается от других. Можно перемещать, увеличивать изображение в окне. Выделение и перемещение кривых происходит точно такое же, как и в случае с обычными объектами. При входе в режим Edit Mode возможна работа с отдельными точками. Не будем повторяться – просто загляните во вторую часть урока (LXF89), где рассказывается про работу со сплайнами.

Задача №3.

Анимация внутренней структуры объекта или самого объекта с учетом внешних воздействий.

Для решения этой задачи имеется несколько путей: либо использование ключевой анимации (с некоторыми нюансами), либо – встроенного физического движка программы. С него-то мы и начнем.

Любой объект в *Blender*, помимо стандартных трехмерных показателей, может иметь и физические, такие как масса, упругость, трение… Все это обеспечивается так называемой системой мягких тел.

Откройте файл **lesson3_begin.blend**, который находится на диске журнала, или же свой проект, оставшийся с прошлого урока. Теперь нашей задачей является создание флага и имитация движения полотнища под воздействием ветра.

Установите 3D-курсор в свободном месте окна Top View (NumPad7) справа от модели пингвина. Древко нашего флага мы будем делать из примитива. Создайте его, нажав пробел и выбрав Add->Mesh->Cylinder. Значение, предлагаемое программой, оставьте без изменений (**vertices=32**). Перейдите в режим Object Mode (Tab) и в окне Transform Properties (клавиша N) введите следующие параметры:

ScaleX=0,100
ScaleY=0,100
ScaleZ=3,500

Переименуйте объект в Staff. Теперь перейдем к созданию полотнища. Для этого идеально подходит примитив Plane (плоскость). Переключите вид на Front View (NumPad1), немного сместите курсор и создайте поверхность (Add->Mesh->Plane). Чтобы придать ему вид флага, растяните его (S) по координате Z до **1,4000**, а затем по X до **2,0000**. Переименуйте полученный объект во Flag.

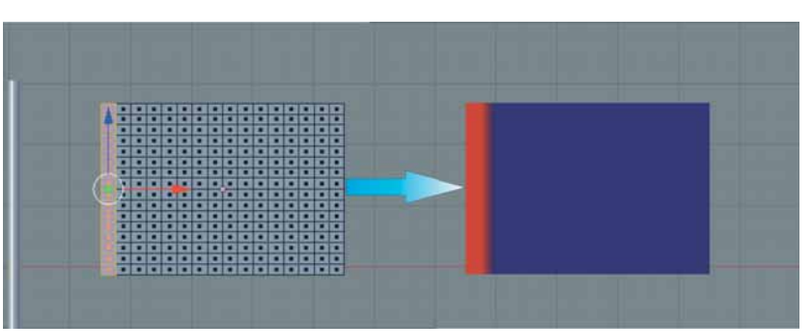
Теперь необходимо разделить объект на дополнительные полигоны. Это можно сделать, перейдя в режим редактирования. Выделите объект (A), нажмите W и выберите пункт меню Subdivide. Проделайте эту операцию еще три раза.

Из подготовительной части осталось только включить сглаживание (F9, Set Smooth). В дальнейшем это нам пригодится.

Итак, система мягких тел в *Blender*'е позволяет реализовать движение вершин объекта в соответствии с законами физики. Ключевое слово в этой фразе – «вершины». Поэтому сейчас нам необходимо подготовить объект к работе с физикой.

Перейдите в окно Front View (NumPad1) и выделите полотнище флага. В режиме редактирования нажмите F9. Инструментарий для работы с вершинами находится в закладке Link and Materials -> Vertex Groups. Проследите, чтобы все вершины флага были выделены и нажмите кнопку New для создания новой группы. Введите в поле Weight значение **0**, переименуйте группу в 1, нажмите Assign. Таким образом, вы подготовите группу вершин, которые в дальнейшем будут обрабатываться физическим движком.

Теперь необходимо отделить от группы некоторое количество вершин для закрепления их на древке; естественно, к ним физика применяться не будет. Для этого снимите общее выделение (A) и отметьте край полотнища толщиной в одну клетку для крепления (**рис. 3а**).



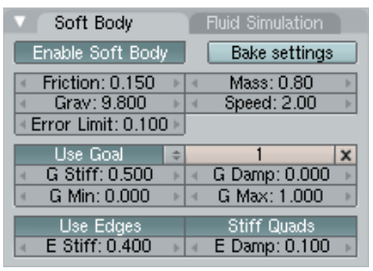
› **Рисунок 3а.**

› **Рисунок 3б.**

Установите параметр Weight равным **1** и повторно выполните при- »

» соединение (Assign). Чтобы удостовериться в правильности наших настроек, можно перейти в режим Weight Paint (находится в том же меню, что и Object Mode, Edit Mode). Картинка должна выглядеть, как на **рисунке 3б**.

Для перехода к настройкам физики нажмите «F7» и активируйте кнопку Physics Buttons. Нас интересует закладка Soft Body. При включении единственной находящейся там кнопки раскроется следующее окно (**рис. 4**).



Физика мягкого тела

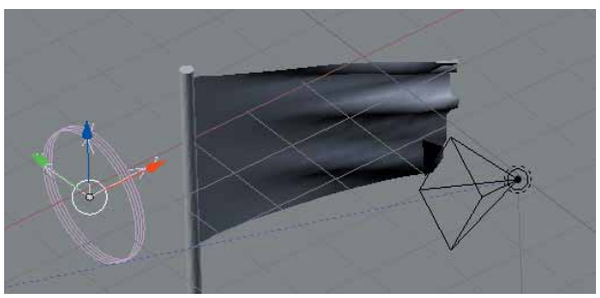
» **Bake settings** Открывает дополнительное окно, в котором можно настроить и выполнить предварительный расчет физики объекта. При общем рендеринге картинки не будет производится вторично.
» **Friction** (Трение) Меньше трения – больше движения, скажем, при использовании ветра.
» **Grav** Гравитация, сила, которая направлена в отрицательном направлении по оси Z.
» **Error Limit** (Лимит ошибок) Определяет размер шагов во время симуляции. Меньшее значение дает большую точность, но повышает время вычисления.
» **Mass** (Масса) Определяет массу вершин. Большая масса замедляет движение, но не для гравитации, где движение есть константа и не зависит от массы.
» **Speed** (Скорость) Контролирует внутреннюю скорость мягкого тела.
» **Apply Deform First** (Сначала применить деформацию) Позволяет применить эффект мягкого тела после деформации объекта Арматурой или Решеткой.
» **Use Goal** (Использовать Цель) Использует для симуляции движение из анимации (Ipo, Deform, Parents, и т.д.). “Goal” (Цель) – это желаемая конечная позиция для вершин, основанных на этой анимации. Достижение “Цели” управляется силой упругости и демпфирования (Damp).
» **Goal** (Цель) Значение “цели” по умолчанию, если не назначена группа вершин (Vertex Group). Если группа вершин назначена и используется, то эта кнопка будет отображать название группы вершин.
» **G Stiff** Пружинная упругость для “цели”. Низкое значение дает больше эластичности, а высокое – больше упругости.
» **G Damp** Коэффициент трения для “цели”. Высокое значение увеличивает демпферный эффект на мягкое тело.
» **G Min/G Max** Когда вы задаете значение воздействия на вершины с помощью окрашивания (используя WeightPaint), то для более точной настройки можно использовать *GMin* и *Gmax*. Самое низкое воздействие (синий цвет) будет соответствовать *GMin*, а самое высокое (красный цвет) - *GMax*.
» **Use Edges** (Использовать ребра) Ребра mesh-объекта могут работать как пружины.
» **Stiff Quads** - Если для граней включена опция quad, то диагональные ребра будут использоваться как пружины. Это предотвращает полное ослабевание граней.
» **E Stiff** - Пружинная упругость для ребер (определяет, насколько они прочные). Низкое значение означает очень малую упругость (очень гибкий материал), а высокое значение соответствует большой жесткости (жесткий материал).
» **E Damp** - Сила трения для граней. Высокое значение снижает трение (т.е. “смазывает” ребра “смазкой”).

Перенесите все значения, представленные на рисунке, в ваш проект. Чтобы указать программе, с каким набором вершин нужно работать, нажмите маленькую кнопочку рядом с Use Goal и выберите группу 1. Для проверки сделанного необходимо перевести курсор в окно проекции и нажать Alt+A. Этим сочетанием клавиш вы запустите анимацию в данном окне. Заметили, что полотнище безвольно колышется? Необходимо добавить ветер, но прежде чем мы сделаем это, ознакомьтесь с настройками Soft Body во врезке «Физика мягкого тела».

Для создания ветра необходимо перейти в окно Top View. Установите курсор слева от флагштока и создайте пустой объект (Add->Empty). Подобного рода объект является лишь служебным каркасом и не отображается при рендеринге. Разверните его так, чтобы ось «Z» смотрела вдоль полотнища, и поменяйте название на Wind. Таким образом мы указываем направление ветра.

Чтобы применить нужный эффект, нажмите F7, активируйте Physics Buttons, в закладке Fields and Deflection -> Fields выберите Wind из выпадающего списка. Данная функция предлагает нам следующие настройки:
» Strength (Сила);
» Fall-off (Затухание) Ослабление силы ветра с расстоянием;
» MaxDist Максимальное область действия эффекта по координате Z.

Поменяйте значение параметра «strength» на **0,8**. Проиграйте анимацию… Вот теперь наш флаг колышется более правдоподобно. Конечно же, не мешало бы сделать направление и силу ветра переменными, но это вы сможете сделать самостоятельно после прочтения данного урока. А пока давайте совместим полотнище с флагштоком (**рис. 5**) и перейдем к следующей части – созданию и анимации земного шара.



Задача №4.

Заставить шарик вращаться и наложить на него текстуру.

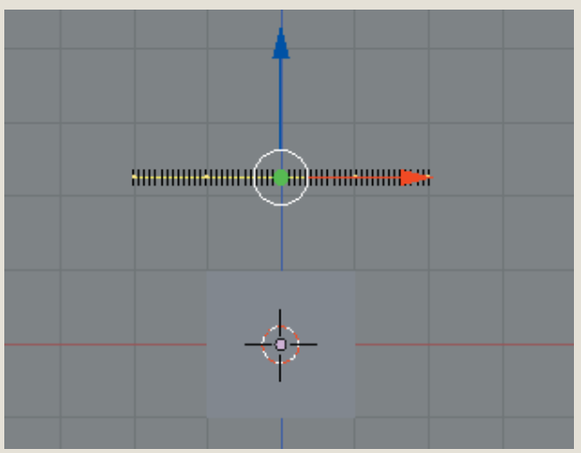
Для сотворения Земли мы воспользуемся ключевой анимацией и редактором IPO Curve. Начнем же, естественно, с создания модели. Установите 3D-курсор в свободной области окна Front View ниже модели пингвина с учетом объемности создаваемого макета. Добавьте в сцену примитив UVSphere с параметрами **64,64**. Отмасштабируйте его по всем осям до **20,000** единиц. Если при этом он перекроет другие объекты – ничего страшного, просто передвиньте его в более подходящее место. Совмещением моделей мы займемся попозже. Переименуйте сферу в Earth и примените сглаживание (Set Smooth).

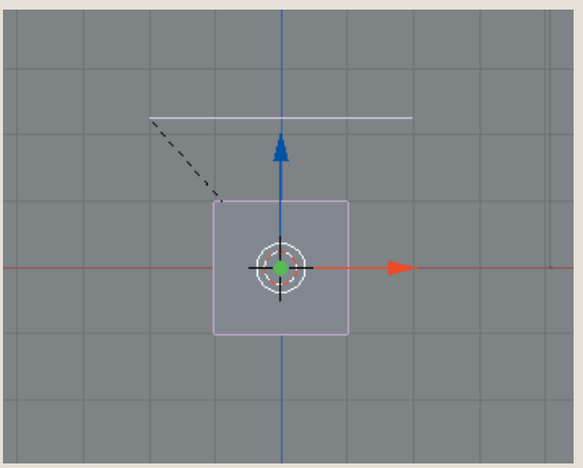
Работу над этой частью анимации лучше всего продолжать, применив схему расположения окон Sr:1 – Animation. Переключите окно с моделями в режим Front View. Выделите шар, нажмите I для вызова меню вставки ключа. Так как предполагается осевое вращение по одной из координат, выберите начальной точкой параметр Rot. Передвиньте анимационный курсор на значение **250** или введите его в соответствующем окне (см. начало урока). Включите запись анимации и поверните объект (R) по координате X на значение **–10,00**. Отключите запись и проиграйте. Если все сделано верно, вы увидите неполный поворот шара по оси X. Посмотрите теперь на окно редактора IPO Curve. Представленная там кривая RotX не является прямой, а значит, анимация будет с ускорением и замедлением (если кривую не видно, необходимо «отъехать» колесиком мыши). Нам это не нужно. «Спрямление»

Комментарий Александра Супрунова

Иногда оказывается, что достичь желаемого результата, используя ключевую анимацию, затруднительно. Тогда на помощь приходит анимация вдоль пути. С ее помощью многие вещи можно сделать значительно проще.

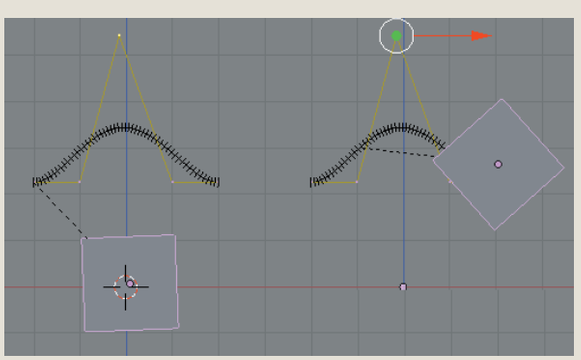
Например, в сцене данного урока Tux гордо шагает по вращающемуся шару, но если бы нам потребовалось также изобразить Луну, делающую оборот вокруг Земли, или надпись “Linux Forever”, тут-то бы и пригодилась анимация вдоль пути.





1 Давайте попрактикуемся. Начните с чистой сцены, на которой есть только куб.
Переключитесь на вид спереди (NumPad1) и добавьте путь: Пробел-> Add->Curve->Path.
Расположите объект Path (он изображен ребристой линией) чуть выше куба.

2 Привяжите анимацию куба к пути. Для этого: перейдите в Object Mode (клавиша Tab), выделите куб правой кнопкой мыши, затем щелкните правой кнопкой мыши по Path, удерживая нажатой клавишу Shift (таким образом вы выделите оба объекта) и нажмите Ctrl+P. В появившемся меню выберите пункт Follow Path – появится пунктирная линий, связывающая куб с Path. Осталось только нажать Alt+A и убедиться, что анимация работает (куб движется вдоль пути). Для завершения анимации нажмите Escape.



3 А теперь самое интересное. Перейдите в Edit Mode (Tab), выделите любой из узлов пути щелчком мыши и тащите его вверх. Завершите перемещение тем же щелчком мыши. Проиграйте изменившуюся анимацию (Alt+A).
Только подумайте, сколько всего интересного можно создать, приложив к этой схеме немного фантазии!

можно выполнить вручную, выделяя каждую точку и работая с ними как с кривыми Безье, или же воспользоваться специальной функцией.

Выделите кривую RotX в режиме Object Mode (правой клавишей мыши или цветной кнопкой у названия), затем вызовите функцию Extrapolation из меню Curve -> ExtendMode. Удалите лишние кривые. Просто выделите их и нажмите клавишу X.

Вот и все, теперь наш шар крутится с постоянной скоростью в нужном направлении.

С анимацией мы закончили, осталось «раскрасить этот мир» и расставить объекты по местам. Начнем, пожалуй, с древка. Так как у нас не стоит задача добиться фотореалистичности, то оставим в покое процедурные карты и текстуры высокого разрешения, и просто создадим материал соответствующей расцветки. Напомню, что для этого нужно перейти в Object Mode, нажать сначала F5, а затем кнопку Materials Buttons и в закладке Links and Pipeline выбрать Add New (см. LXF89). Поменяйте значение RGB на: **0,500**; **0,300**; **0**.

Помимо стандартной расцветки материала, *Blender* позволяет использовать многоуровневые наложения в виде слоев. В новой версии программы вы можете использовать до 10 текстур в одном материале. Не стоит думать, что под термином «текстура» скрывается возможность использования лишь сторонних картинок! *Blender* имеет ряд предустановленных, входящих в стандартную комплектацию процедурных карт, которые вы с успехом можете использовать в своих работах. Процедурными эти типы текстур называются потому, что для создания их используются специальные программные модули. С их помощью вы можете сгенерировать вполне реальные облака, дерево, ландшафт, шершавые поверхности и многое другое. Слои могут отображаться одновременно при помощи механизма смешивания. Благодаря этому можно »

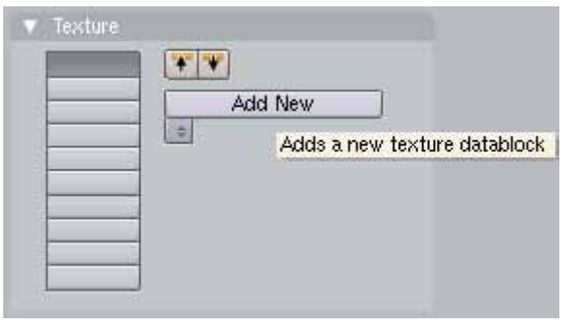


Рисунок 6.

добиться весьма интересных эффектов, но это уже отдельная история. Мы же пока воспользуемся наложением внешней текстуры на объекты.

В директории с файлами к этому уроку на **LXF**DVD вы найдете две текстуры для отображения земной поверхности и флага. Создайте новый материал для флага и в закладке Texture нажмите Add New. Так в текущий материал добавляются новые слои (**рис. 6**).

Настройка созданной текстуры производится в панели, вызываемой клавишей F6. Выберите из выпадающего списка Texture Type тип Image. Осталось только загрузить нужный файл в проект, нажав Load Image (файл **flag_tex.jpg**). Проделайте то же самое для шара, только текстура в данном случае будет называться **earth.jpg**.

Все необходимые настройки загруженных картинок находятся в данной закладке. Вы можете указать смещение текстуры относительно объекта, воспользовавшись значениями MinX, MaxX, MinY, MaxY, или развернуть ее на 90 градусов функцией Rot9. Если тип картинки поддерживает канал прозрачности (альфа-канал), его можно включить кнопкой Use Alpha. Для повтора узора на поверхности имеется кнопка Repeat, количество повторов можно регулировать значениями Xrepeat, Yrepeat. Кроме того, поддерживается настройка анимации текстуры на объекте.

Вот так, очень просто, можно раскрасить безликие примитивы.

Задача №5.

Разместить объекты на сцене.

Для завершения работы над нашей сценой необходимо разместить объекты в нужных местах. Задача эта непростая, если перемещать каждый объект вручную. Очень часто при построении сложной сцены и перемещении моделей в ней теряются некоторые компоненты. Поэтому важна строгая иерархия. *Blender* поддерживает создание групп со сложной подчинительной структурой, состоящей из материнского объекта и нескольких дочерних. Схватившись за родительский объект, можно перетянуть всю нижестоящую иерархию.

Выделите весь флаг таким образом, чтобы шток был отмечен последним (не забудьте и про «ветер»!). По умолчанию, родительским объявляется именно последний объект. Нажмите Ctrl+G – Add New Group для создания новой группы и переименуйте ее во Flag. Затем создайте иерархию комбинацией Ctrl+P – Make Parent. Вот и все, для проверки попробуйте перетянуть группу, ухватившись за шток флага. Если все сделано правильно – перенесется вся группа.

Следующим на очереди у нас идет пингвин. Проделайте ту же операцию, в качестве родительского объекта выберите тело (Body). Новую группу можно назвать Tux.

Итак, у нас имеются группы и объекты: Tux, Flag, Lamp, Earth, Camera. Удобнее всего выполнить нашу задачу в SR:1 - Animation. Окно слева вверху демонстрирует иерархическую структуру сцены. Теперь, выделяйте нужный объект или группу в этом окне, переносите курсор

T.P.	Earth	Tux (Body)	Flag (Staff)	Camera	Lamp
LocX	-0,600	-0,600	1,150	-0,600	-0,600
LocY	0,800	1,000	1,030	-22,600	-22,600
LocZ	-17,000	4,000	6,500	2,750	2,750
RotX	--	90,0	0,0	--	--
RotY	--	0,0	0,0	--	--
RotZ	--	0,0	20,0	--	--

Рисунок 7.

в окно View и вводите значения из таблицы (**рис. 7**) в окно Transform Properties (клавиша N).

Выполните наводку камеры и лампы на объект Body. Для этого выделите камеру, затем туловище, и нажмите Ctrl+T. То же самое проделайте и для лампы.

Вот мы и подошли к завершению этого урока. Осталось только разобраться с рендером анимации и обработать первый свой ролик из-под *Blender*.

При нажатии F10 вы попадете в окно настройки рендера (**рис. 8**). Обратите внимание на закладку Format. Здесь вы можете выбрать один из заготовленных стандартов, либо ввести удобные для себя значения. Чуть ниже таблицы разрешений из выпадающего меню можно выбрать необходимый вид сохранения результата, настроить кодек, установить количество кадров. Для демонстрации результата советую использовать предустановленный тип Preview, как наиболее скоростной из имеющихся.

Рисунок 8.

В панели Output необходимо указать путь для сохранения результата, а закладка Anim позволяет выполнить саму обработку (кнопка Anim), проиграть последнюю анимацию (Play) встроенным проигрывателем программы, настроить начало и конец анимации для обработки (значения Start, End). Выполните все выше сказанное и… запаситесь терпением, обработка может занять значительное время.

Впереди у нас остался последний, заключительный урок из серии «*Blender* для начинающих», где мы познакомимся со скелетной анимацией, поговорим о правильном освещении, окончательно доработаем сцену. LXF

Работа над ошибками

В прошлом номере нашего журнала мы по ошибке указали в качестве автора комментариев к учебнику *Blender* Сергея Супрунова.

Редакция LinuxFormat приносит извинения своим авторам и читателям журнала.
Соавтором **Андрея Прахова** в цикле статей о пакете *Blender* и автором комментариев в прошлом и нынешнем номерах является **Александр Супрунов.**

Через месяц Мы обсудим скелетную анимацию и подберем для нашей сцены выгодное освещение.

Ответы

Есть вопрос по Open Source? Пишите нам по адресу: answers@linuxformat.ru

В этом месяце мы отвечаем на вопросы по:

1. Разрешениям экрана
2. Заданиям cron
3. CUPS
4. NdisWrapper
5. Мусору в /home
6. Загрузке с USB
7. Mepis OnTheGo
8. Moodle
9. grep и sed
10. VNC
11. Драйверам Nvidia
12. Раскладке клавиатуры
13. Сетевому чату
14. Совместному использованию соединения
★ Очистке звука
★ Zen-updater'у

1 Куда деваются кнопки?

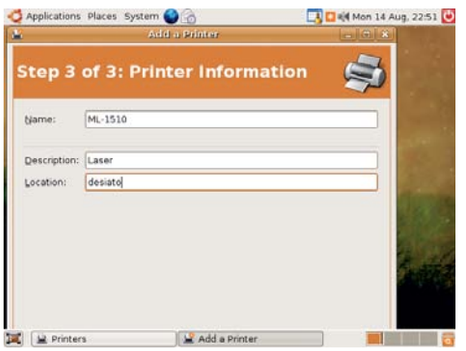
Я работаю в Ubuntu 6.06 с диска к LXF83 на Compaq Presario SR1720NX, и я новичок. Когда я пытаюсь добавить в систему принтер Epson Stylus CX4800 с помощью Gnome CUPS, низ экрана, где должны быть кнопки Отмена, Назад, Вперед и Применить, пропадает. Я могу нажать на кнопку Enter вместо Вперед, но на экране 3 я не нашел способа перейти на кнопку Применить.

Jim Laprad [Джим Лапрад]

Здесь, похоже, проблема с разрешением экрана. Не получив точную информацию от монитора о допустимых разрешениях, установщик Ubuntu выбирает безопасное 640x480, этого слишком мало, чтобы вместить целиком окно добавления принтера. Быстро помочь горю можно, зажав клавишу Alt и перетащив окно левой кнопкой мыши так, чтобы кнопки внизу стали видны. Удержание Alt позволяет перетаскивать окно за любое место, а не только за заголовок. Это поможет вам добавить принтер, но не решит проблему как таковую.

Для смены разрешения монитора на более подходящее воспользуйтесь пунктом Настройки > Разрешение экрана из меню Система. Там перечислены все подходящие разрешения для ваших видеокарты и монитора. Если обнаруживается только 640x480, значит, ваше оборудование не было распознано при установке. Менеджер устройств (Система -> Администрирование) покажет, правильно ли была определена видеокарта. На вашей машине это, по идее, ATI Radeon XPress 200 IGP. Для смены настроек видеокарты и монитора запустите *dexconf*, он определит оборудование и запишет все в конфигурационный файл. Предварительно хорошо бы сделать резервную копию текущих настроек:

```
cp /etc/X11/xorg.conf ~
sudo dexconf
```



Нажав клавишу Alt, можно перетаскивать мышью окна, не влезающие в экран.

Будет сгенерирован новый файл **/etc/X11/xorg.conf**, а его резервная копия останется в Вашем домашнем каталоге. После этого перезапустите X-сервер. Это можно сделать из командной строки, но если вы новичок, проще перезагрузить компьютер.

Если по-прежнему будет доступно лишь 640x480, придется вручную редактировать **xorg.conf**. Не видя Вашу конфигурацию, нельзя точно сказать, что именно потребуется изменить. Если вы зашли так далеко и не получили ничего, кроме 640x480, рекомендую вам спросить совета на нашем форуме **www.linuxformat.co.uk**, приложив содержимое вашего **/etc/X11/xorg.conf**, вывод **lspci -v** и параметры вашего монитора. НБ

2 Задачи времени

У меня небольшая проблема с задачами *Cron* на SUSE 9.3. Скрипты, помещенные в директории **cron.hourly**, **cron.daily** и **cron.weekly**, работают отлично, но как мне контролировать запуск файлов из этих директорий? Можно ли выполнять какие-то еженедельные задания в субботу, а какие-то – в пятницу, а ежедневные – вечером или в полночь? Хотел проследить, как это работает, но ясности не добился.

Единственно, что я нашел, это задание, записанное в *crontabs*, которое запускается каждые несколько минут и работает с заданиями из директорий **cron.***. Скрипт смотрит, какие из заданий должны выполняться в данный момент, и запускает их.

Марк Бесяда [Mark Biesiada]

О SUSE 9.3 делает это не совсем так, как другие дистрибутивы. Вместо запуска содержимого этих директорий в определенное время, она запускает их согласно времени последнего исполнения. Вы почти добрались до этого сами: одна строка в **/etc/crontab** вызывает скрипт *run-crons* каждые 15 минут. Тот смотрит на файлы-маркеры каждой директории **/etc/cron.***, находящиеся в **/var/spool/cron/lastrun**. Если маркеру больше, чем час/день/неделя, он запускает скрипты в директории и обновляет маркер. Если маркера нет, он запускает скрипты и создает его.

Наши эксперты

Мы найдем эксперта на любой вопрос! Вы получите ответ на все: от проблем с установкой или модемом до сетевого администрирования; главное – спросить!

Нейл Ботвик
Владелец ISP и экс-редактор дисков для нашего журнала, Нейл считает, что в Linux он от скуки на все руки.

Майк Сондерс
Майк был одним из создателей прототипа LXF – Linux Answers. Его специальности – программирование, оконные менеджеры, скрипты инициализации и SNES.

Стефан Лукас
У Стефана шесть лет администраторского стажа. Теперь он администратор L2 Linux в Rackspace, в его коллекции серверов есть все от Raq3 до Sun Netra X1.

Ник Вейч
В свободное от исчеркивания текстов красными чернилами время Ник возится с Linux-графикой и 3D-приложениями; он у нас отвечает за простые вопросы!

Петр Семилетов
Разработчик текстового редактора TEA, режиссер-любитель, поможет вам разобраться в мультимедийных вопросах.

Валентин Синицын
В свободное от работы время редактор нашего журнала разрабатывает KNetworkManager и другие открытые приложения. Он с радостью поможет вам в вопросах использования Linux на рабочем столе.

КУДА ПОСЫЛАТЬ ВОПРОСЫ:
Пишите нам по адресу: **answers@linuxformat.ru** или спрашивайте на форуме: **www.linuxforum.ru**

» Выходит, что вместо запуска ежедневных скриптов в 4:30, когда загрузка системы минимальна, они запускаются через день после последнего выполнения. Вы можете указать нужное время запуска, обновив временную метку файлов в **/var/spool/cron/lastrun**. Следующий скрипт поменяет ее на 4:30 для ежедневных, еженедельных и ежемесячных маркеров.

```
#!/bin/sh
cd /var/spool/cron/lastrun
for i in daily weekly monthly
do
        if [ -f cron.$i ]
        then
                touch -t $(date -r cron.$i+%Y%m%d)
0430 cron.$i
        fi
done
```

Скрипт использует команду date для извлечения текущей даты файла в формате ГГГГММДД, добавляет нужное время (**0430**) и передает это команде touch, которая и обновляет временную метку. Тем же способом можно поменять день выполнения еженедельных скриптов.

Учтите, что если вы измените временной штамп, а в 4:30 задача не выполнится (например, если ваша машина будет выключена), временная метка перепишется на то время, когда задача сработает, и вам придется заново изменять ее. Этот процесс можно автоматизировать, создав отдельную задачу в **/etc/crontab** и запуская ее каждый день в 4:00. **НВ**

3 Печать в Gentoo

Установка Gentoo прошла неожиданно гладко, но заставить свой USB-принтер печатать я не могу. Тщательно следовал инструкциям из Gentoo Printing Guide, трижды проверил конфигурацию моего ядра и абсолютно уверен, что включал все, что нужно. Пробовал собирать ядро с USB-опциями в виде модулей и вкомпилированными в ядро. Ничем не помог *genkernel*, так что вряд ли дело в ядре.

Принтер – Samsung ML1210. Он хоть и снят с производства, но мне годится, и всегда работал в Linux хорошо, да и сейчас отлично печатает в Ubuntu Edgy на той же машине и на том же USB-порту, то есть это не аппаратная проблема и не CUPS.

Если я открываю *менеджер печати Gnome*, мастер предлагает мне те же драйверы CUPS, что и в других дистрибутивах, но при попытке печати тестовой страницы ничего не происходит. То же самое и в *OpenOffice.org*. OOo думает, что распечатал документ, а документа нету.

Вот вывод *lsusb*:

Bus 002 Device 003: ID 04e8:300c Samsung Electronics Co., Ltd ML-1210 Printer

Я проверил **/var/log/cups/error_log** и не увидел ничего неблагоприятного.

«Пятнистый кот», с форума LXF

При определении проблем CUPS первым делом нужно повысить уровень ведения журнала. Измените LogLevel в **/etc/cups/cupsd.conf** с ‘info’ на ‘debug’ и перезапустите CUPS.

В вашем случае улика обнаруживается в приведенных Вами записях. Вы используете *GPL Ghostscript*, где нет должной поддержки драйверов для GDI-принтеров (они же Win-принтеры) вроде Вашего Samsung. Так что удалите **ghostscript-gpl** и поставьте **ghostscript-esp**, дающий лучшую поддержку принтеров:

```
emerge --unmerge ghostscript-gpl
emerge --oneshot ghostscript-esp
```

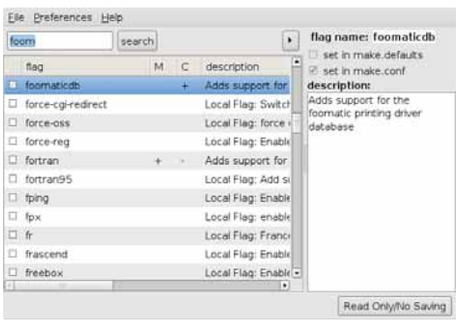
Вероятно, Вам нужен еще пакет **openslp**, хотя он и считается опциональной зависимостью CUPS. SLP (Service Locator Protocol) пригодится и другим программам, так что добавьте его в USE-флаги в **/etc/make.conf**. Еще стоит добавить **foomaticdb**: напрямую он на CUPS не влияет, но может повысить уровень поддержки принтеров в приложениях.

Теперь пересоберите все пакеты, для которых изменились флаги, включая CUPS:

```
emerge --newuse --deep --verbose --ask world
```

Отобразится список всех пересобранных пакетов, включая CUPS и OpenSLP. Нажмите Enter для их установки и перезапустите CUPS, когда сборка завершится.

USE-флаги – важная часть Gentoo, все они описаны в **/usr/portage/profiles/use.desc** и **/usr/portage/profiles/use.local.desc**. Еще можно установить Profuse и искать, просматривать и устанавливать их из графического интерфейса. **НБ**



› Оптимизируете Gentoo? Установите USE-флаги через дружелюбный интерфейс *Profuse*.

4 Бедствия WiFi

Вот думаю, не поможете ли вы мне с проблемой с WiFi/*NdisWrapper*. Я пытаюсь заставить работать карту Belkin под *NdisWrapper* с драйвером rt2500 для Windows XP. Онлайн-инструкции превосходны; я распознал свою карту и понял, какой драйвер мне нужен. Установил *NdisWrapper* и этот драйвер, и когда я набираю **ndiswrapper-l**, видно, что драйвер установлен и оборудование налицо.

После этого я загрузил *NdisWrapper* в ядро через **modprobe**, настроил сеть, и все отлично работало. А потом перезагрузился, и настройки, естественно, пропали. И теперь я не могу добиться работы. *NdisWrapper* по-прежнему показывает, что все в порядке, но индикаторы на карте отключены, и когда я пытаюсь получить ip по DHCP, выдается сообщение «отсутствует связь, проверьте провод».

Я перезапускал **modprobe ndiswrapper**, вынимал и вставлял карту заново, но индикаторы на ней все равно не горят.

Эндрью Вуд [Andrew Wood]

Подобная проблема нередка для *NdisWrapper*, но вас она не должна волновать. Для карт на основе rt2500 незачем использовать *NdisWrapper*: это имеет смысл лишь при отсутствии подходящего Linux-драйвера (выполнение Windows-

Краткая справка

Директория proc

Виртуальный снимок текущего состояния системы.

Вы когда-нибудь просматривали свою файловую систему, исследуя, куда девалось место на жестком диске, который вы считали бездонным? Тогда вы могли наткнуться на директорию **proc**, занимающую больше гигабайта и содержащую тысячи файлов, многие из которых на вид пусты, и задуматься: а не удалить ли их?

Уж извините, удалить их нельзя. По счастью, они фактически не занимают места на диске, поскольку **/proc** – виртуальная файловая система. Это не коллекция реальных файлов, находящихся на Вашем жестком диске, это всего лишь снимок текущего состояния ядра, представленный в виде файлов, чтобы с ним было удобно работать.

Как вы видите, **/proc** содержит много «файлов»; на нашем компьютере их 7109, и каждый содержит некую информацию. Например, набрав cat **/proc/version**, вы узнаете больше о ядре вашей системы; а cat **/proc/**cpuinfo выведет детальную информацию о процессоре. Директории **ide**, **scsi**, **bus/usb**, **bus/pci** и **bluetooth** содержат данные о соответствующих устройствах.

Вы также можете заметить директории, названные цифрами. Каждая из них соответствует работающему процессу, идентификатор которого и является именем директории. В каждой из них находится детальная информация о процессе: используемая память, аргументы командной строки, с которыми он был запущен, и многое другое.

```
[nelz@nactar ~]$ cat /proc/cpuinfo
processor       : 0
vendor_id       : AuthenticAMD
cpu family      : 15
model           : 4
model name      : AMD Athlon(tm) 64 Processor 3200+
stepping        : 10
cpu MHz         : 2000.143
cache size      : 1024 KB
fpu             : yes
fpu_exception   : yes
cpuid level     : 1
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca
t lm 3dnowext 3dnow
bogomips        : 4002.32
TLB size        : 1024 4K pages
clflush size    : 64
cache_alignment : 64
address sizes   : 40 bits physical, 48 bits virtual
power management: ts fid vid ttp
[nelz@nactar ~]$
```

› Исследуем внутренности системы в /proc.

Многое из **/proc** доступно только для чтения, но в некоторые файлы вы можете писать. Например, **/proc/sys/vm/swappiness** контролирует, насколько агрессивно система будет сбрасывать страницы в раздел подкачки – чем больше номер, тем больше сбросов. Поменять значение можно командой *echo*:

```
echo 30 >/proc/sys/vm/swappiness
```

Подробные сведения о **/proc** ищите в **/usr/src/linux/Documentation/filesystems/proc.txt**.

кода от имени root – плохая идея, если этого можно избежать).

Linux-драйверы для чипсета rt2500 доступны на http://rt2x00.serialmonkey.com и http://sourceforge.net/projects/rt2400. Не беспокойтесь насчет 2400 в имени: один и тот же проект занимается драйверами как для rt2400, так и для rt2500. Это полуофициальные драйверы, на базе закрытых драйверах от Ralink, которые впоследствии вышли под GPL. Кроме самих драйверов, проект занимается разработкой графических утилит для настройки карт и сканирования.

Одни дистрибутивы, например, Debian, включают драйверы в свои репозитории, для других вам придется собирать их из исходных кодов. Не зная, что у вас за дистрибутив, сложно дать конкретные инструкции по установке, но если вы будете собирать драйверы из исходников, потребуется также исходный код ядра. Обычно он находится в пакетах **kernel-sources**, **linux-sources** или **kernel-devel**.

Удостоверьтесь, что вы установили исходники той же версии, что и ваше ядро. Как и со всеми внешними модулями ядра, при обновлении ядра нужно будет обновить и модули. Поскольку в это время у вас может не быть доступа в Интернет, рекомендую сохранить копию исходных текстов или инсталляционный пакет.

Если вы все-таки хотите использовать *NdisWrapper*, нужно будет запустить **ndiswrapper -m** для настройки псевдонима wlan0. Это заставит *NdisWrapper* загружать модуль и драйвер. **MC**

› Советы по графическим картам приводятся на стр. 26.

5 Домашняя уборка

В Понаставив всяких дистрибутивов, я замусорил директорию **/home/username**, а в главном меню теперь полно неработающих пунктов. Как удалить эти мертвые души и отсортировать файлы по расширениям, поместив, например, все GIF, PNG и JPEG-файлы в одну директорию? Также хотелось бы разобраться с дубликатами файлов. Я готовлюсь к переустановке Ubuntu.

AJB2K3, с форума LXF

О Я бы переименовал **/home/username** перед установкой, а затем скопировал только нужные файлы. Возможно, это будет проще, чем очищать домашнюю директорию от обломков других систем. Лучшая программа для удаления ненужных файлов, которую я нашел, называется *KleanSweep* (http://linux.bydg.org/~yogin).

С поиском дубликатов лучше всего справляется Fdupes с (http://netdial.caribe.net/~adrian2/fdupes.html). Ее можно использовать так:

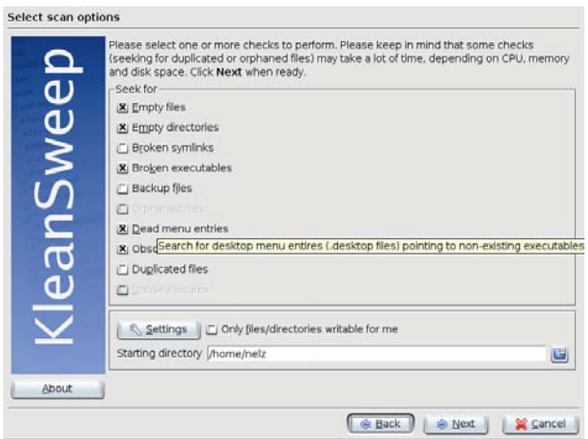
```
fdupes --recurse ~
fdupes --recurse --omitfirst ~ | xargs rm
```

Первая команда покажет все найденные в домашнем каталоге дубликаты, вторая удалит все, кроме первого – пользуйтесь с осторожностью!

Сортировать файлы по имени лучше всего командой find. Вы можете переместить все изображения в один каталог так:

```
mkdir pics
find ~ -iname `*.jpg' -o -iname `*.png' -o -iname
`*.gif' -exec mv "{}" pics `;'
```

НБ



› Время весенней чистки машины? *KleanSweep* избавит вас от ненужного мусора.

6 Linux на брелке

В У меня есть USB-брелок на 2 ГБ, на нем установлен Slax Popcorn Edition. Я запросто загружаю с него компьютер и сохраняю изменения в системе. Время от времени я сталкиваюсь с тем, что мне нужно загружаться с брелка, но перезагрузка невозможна, а так как машина «жестко» сконфигурирована, *Qemu* воспользоваться нельзя.

Я пытался найти решение этой проблемы. Я пробовал *VMPlayer*, *Qemu* и *Moka*, и все без толку: чего-нибудь да не хватает. Подходящим решением было бы установить *VMPlayer* прямо на брелок и туда же записать образ системы, но я не смог найти способа это сделать.

Есть ли на рынке решение, позволяющее запускать свою собственную ОС с USB-брелка независимо от установленной на компьютере системы?

Virtaava

О Есть несколько причин, по которым загрузка с брелка может быть невозможна на некотором оборудовании. На некоторых компьютерах нет опции загрузки с USB-устройств, но к счастью, сейчас это уже редкость. Другой возможный сценарий – хозя- »

Часто задаваемые вопросы

Пользователи и суперпользователи

Считайте root своим суперсильным виртуальным другом.

» Я только что впервые установил Linux. Зачем мне создавать двух пользователей?

Один из них будет root, суперпользователь, только для администрирования. Другой – обычный пользователь для повседневной работы.

» Почему я не могу администрировать систему сам?

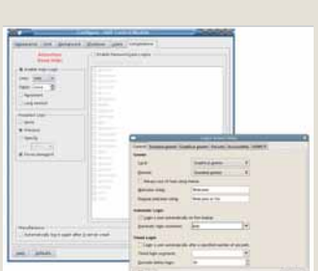
Linux использует права доступа, установленные на каждую директорию и на каждый файл в системе. Каждый пользователь имеет свою домашнюю директорию, где он может создавать и удалять файлы с любым содержанием. Вне ее каждый пользователь имеет право читать файлы и запускать программы, но не изменять и удалять их. Это обеспечивает надежность и безопасность. Не только вы, но даже вредоносное приложение к электронному письму не сможет повредить системные файлы: прав не хватит! Когда вам действительно потребуются права администратора (например, при установке программы), вы можете переключиться на него и работать до тех пор, пока требуются права root.

» Значит, нужно выходить и заходить в систему каждый раз, когда потребуется установить программу? Как это нудно.

Нет, вы не должны ни выходить из системы, ни заходить как root (некоторые дистрибутивы даже предотвращают это). вам нужно только запускать отдельные программы с правами root. Многие программы настроены на это. Если вы попробуете запустить YaST в SUSE или Mandriva Control Center в Mandriva, у вас спросят пароль root. Для запуска других программ, откройте терминал, наберите su, введите пароль и запустите нужную программу, введя ее имя.

» Если моим компьютером пользуюсь я один, должен ли я каждый раз заходить в систему?

Большинство дистрибутивов сейчас имеют опцию автоматического входа под любым пользователем (но не root!), и это удобно, если компьютером пользуетесь только вы один. В рабочем столе KDE опцию можно установить, запустив Центр Управления KDE > Администрирование > Экран входа в систему. В Gnome – Рабочий стол > Система > Окно входа в систему (включите автовход). Но подумайте дважды перед включением этой опции на ноутбуке. Если он будет потерян или украден, любой сможет получить доступ к вашей почте и конфиденциальной информации. Пароли дают некоторую защиту от этого.



› Автоматический вход имеется и в KDE, и в Gnome.

» ин настроил BIOS своего компьютера так, чтобы загрузка с USB была невозможна. Попытка снять это ограничение как минимум некрасива (а то и незаконна), если только вы не получите разрешение хозяина. Если оно у Вас есть, Вы можете использовать загрузочный компакт-диск для передачи управления USB-устройству; на сайте Slax (www.slax.org) есть такой образ.

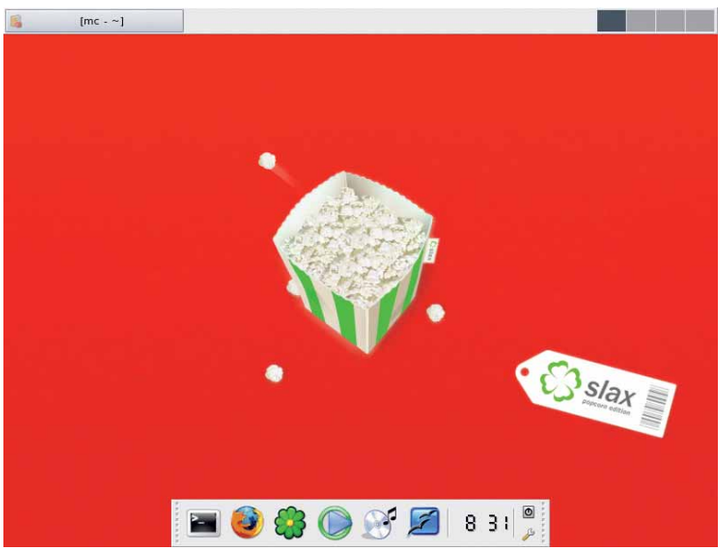
Другое препятствие загрузки с USB-устройства – как минимум три ее способа. Устройство может загружаться как дискета, ZIP и жесткий диск. Установщик Slax на USB, видимо, использует первую опцию.

Не все BIOS'ы умеют загружать систему со всех этих типов носителей, так что может понадобиться больше одного брелка. Damn Small Linux загружается с брелка как USB-ZIP и USB-HDD. Например, на моем ноутбуке Slax загружаться отказался, а DSL спокойно загрузился с той же «флэшки».

Некоторые компьютеры не в состоянии загружаться с USB-брелка, если загрузочный раздел больше 256 МБ, так что Вы должны разбить свой брелок на два раздела – один для системы (256 МБ), другой – для данных.

Ваша идея с VMWare весьма изобретательна, поскольку позволяет избежать перезагрузки, но некоторые файлы *VMPlayer* должны быть обязательно установлены в гостевой ОС. *Moka* выглядит более подходящим решением, но она работает с временными файлами гостевой системы, и ее надо запускать от имени администратора.

Если конфигурация компьютера препятствует загрузке с USB, Вы должны принять это или попросить владельца сменить его. Если таких ограничений нет, попробуйте другой дистрибутив. Mandriva недавно анонсировала Mandriva Flash, полноценную настольную систему на 2 ГБ флэш-диске. Я его пока не пробовал, но Вы можете найти информацию о нем на www.linuxcenter.ru/shop/distros/linux-distros/Mandrava-Flash-2GB-USB. НБ



› **Slax Popcorn – миловидный дистрибутив, но с USB-брелка загружается не без проблем. Проверьте настройку BIOS.**

7 OnTheGo... заглох

В Хотел создать OnTheGo-диск из Live SimplyMepis 6.0, но окно выбора дисков не содержит ни одного пункта. Я пробовал:

1. Загрузить машину с вставленным USB-брелком, затем смонтировать его.
2. Загружать систему без «флэшки», затем вставлять ее и монтировать
3. Заходить в систему как demo и как root.
4. Вставлять «флэшки» Advent на 2 ГБ и Huke на 512 МБ.

Я знаю, что диск нормально монтировался, поскольку я сохранял на нем файлы – выделял несколько изображений и перетаскивал их на него, и все они оставались на диске после перезагрузки. Моему компьютеру около шести лет; это Pentium 3 с установленной Windows 98 и картой расширения PCI-USB2. Мой опыт в Linux ограничивается «живыми» дистрибутивами с журнальных дисков за последние месяцы. Как новичок в Linux, я теряюсь и не знаю, что еще придумать.

Д.Томпсон [D.Thompson]

О Для создания OnTheGo Вы должны войти в систему как root, и USB-устройство должно быть не смонтировано. После авторизации, вставьте устройство. Если диалоговое окно KDE спросит, что делать, отвечайте «Ничего». Если диск монтируется автоматически, размонтируйте его, запустив *KwikDisk* или набрав **umount /dev/sda1** в терминале. Не пользуйтесь опцией безопасного удаления из контекстного меню иконки диска на рабочем столе: она удалит файл устройства в **/dev**, сделав его недоступным для установщика.

Теперь запустите *Mepis Utilites*, выберите опцию создания OnTheGo-диска, и ваш диск должен стать доступным, скорее всего как **sda**. По завершении процесса, удалите USB-диск (размонтировать его нет нужды) и выберите «Завершение сеанса» из меню K. Когда появится экран входа в систему, подключите USB-диск, подождите 10 секунд, пока он определится, и зайдите в систему под именем onthego (пароль onthego). Если вы создали OnTheGo с шифрованием, о пароле вас спросят позже.

OnTheGo-диск хранит только ваши данные, которые могут быть зашифрованы; это не избавляет вас от загрузки с компакт-диска Mepis. С другой стороны, в этом случае не будет проблем с загрузкой с USB-устройства, описанных выше, и вы можете скопировать файл **.onthego.iso** на другой диск, если пожелаете. НБ

8 Linux-планы

В Мне хотелось бы использовать *Moodle*, систему управления курсами, с web-хостингом на базе Linux, *Apache* и *MySQL*. На каком Linux это

Вопрос-победитель (русская версия)

Алексей Николаевич получает подарочный сертификат на 1000 рублей от интернет-магазина LinuxCenter.Ru! Просим победителя выйти на связь с редакцией: info@linuxformat.ru

Операция «Чистый звук»

В У меня возникла следующая проблема. Мои родственники недавно отмечали юбилей и засняли весь праздник на любительскую MiniDV-камеру.

Знакомые сделали им по записанным материалам DVD-диск с простенькими эффектами, меню и т.д. Все получилось хорошо, но вот звук остался необработанным и на записи очень много шума. Меня попросили, по возможности, убрать этот шум. Я записал звуковую дорожку на жесткий диск с помощью mencoder, затем загрузился с *LiveDVD Dynebolic* (с диска из приложения к **LXF**) и попытался удалить из записи лишние шумы при помощи инструмента *Remove Noise* редактора *Audiacity*. Вроде бы получилось неплохо. Теперь у меня возникла проблема – как наложить очищенную звуковую дорожку обратно на исходный DVD? Желательно чтобы при этом сохранилось и меню, которое было на этом диске.

Маслий А.Н.

О Ответ разложим на пункты. Просто так взять да и заменить звуковую дорожку на DVD не получится. DVD имеет четкую файловую структуру, где видео и звук хранится в VOB-файлах, плюс к ним есть разные служебные файлы. Для записи DVD-диска необходимо подготовить видеоматериал в формате MPEG2 и с помощью некой утилиты (например, *QDVDAuthor* – http://qdvdauthor.sourceforge.net) записать MPEG2 таким образом, чтобы файлы MPEG2 (или один большой файл) были разбиты на файлы VOB и размещены на своих местах в структуре. Однако, вы хотите заменить звуковую дорожку. В таком случае вам нужен исходный MPEG2-файл.

Допустим, он называется **source.mpeg**. Теперь нам нужна дорожка в формате PCM или AC3. Пусть будет AC3 (в этот формат можно перевести данные, используя ту же Cinelerra). Файл назовем **sound.ac3**. Не забудьте, что звук должен быть с частотой оцифровки 48 КГц. Теперь совмещаем видео- и звуковой потоки с помощью утилиты *mplex* из состава **mjpegtools/Cinelerra**, примерно (повторяю - примерно) вот такой командой:

```
mplex -o out.mpeg --format 9
source.mpg sound.ac3
```

В итоге получаем в файл **out.mpeg**, который можно вставить в проект *QDVDAuthor*. В *QDVDAuthor* же можно сделать и меню. Чтобы сохранить те меню, которые были на исходном DVD, надо спроектировать весь DVD в той же программе. ПС

лучше всего сделать? *Moodle* допускает установку на Windows, но мне все-таки хотелось бы сделать все по-правильному.

Ли [Lee]

О С какой версии Linux лучше начать? Я думаю, это зависит от Ваших предпочтений. Как пользователь Debian, я предложил бы этот дистрибутив, он стабилен и прост в установке. Если Вы новичок, я бы посоветовал Ubuntu (он основан на Debian). Последняя серверная версия – Ubuntu 6.06 LTS. Я имею опыт с Debian и могу сказать, что установка серверной части займет не более получаса с момента вставки компакт-диска. Пакетная система Debian очень проста в использовании, поможет команда *dselect*. **СЛ**

Вопрос-победитель (английская версия)

В поисках дзэн-тишины

В Может, это звучит странно, но мне хотелось бы, чтобы Zen-updater не запускался, когда пользователь входит в систему. Подскажите, пожалуйста, как это сделать.

Мне это нужно потому, что при входе пользователя через домен, Zen рушится, выдавая сообщение об ошибке. По-моему, это из-за того, что доменные пользователи не имеют локального ID и не могут быть обнаружены в системе. И еще, пользователям ведь не нужно производить обновление системы.

Майк [Mike], с форума LXF

О Решение очень простое. В директории /etc/xdg/autostart есть файл zen-updater-auto.desktop. Вам нужно открыть его в любимом текстовом редакторе и закомментировать строку Icon=zen-icon. После этого нужно перезапустить Zen-updater. СЛ

9 Групповое редактирование

В Мне нужно через *grep* найти в файле строки, содержащие некоторую подстроку, и целиком удалить их. Причем проделать это со множеством файлов. Можете помочь?

Гоутхам Вутхаркар [Goutham Vutharkar]

О Можно использовать *grep* для этого: **grep -v string file** выведет все строки, не содержащие подстроки **string**. Но для группового редактирования больше подходит *sed*.

```
sed --in-place `/some string/d' myfile
```

удалит все строки, содержащие 'some string'. Для обработки множества файлов Вам нужно будет воспользоваться циклом *for* (или утилитой *find*), поскольку опция **--in-place** в *sed* работает только с одиночными файлами. Подойдет одна из этих команд:

```
for f in *.txt; do sed --in-place `/some string/d' "$f"; done
```

```
find -name `*.txt' -exec sed --in-place=.bak `/some string/d' "{}" `;'
```

Добавление **=.bak** в последнем примере заставит sed сохранить резервную копию перед редактированием файла. **НБ**

10 VNC, пожалуйста!

В Я соединяюсь со своим домашним сервером с помощью VNC (не через SSH!). Однако он не отображает KDE-экран входа в систему, и я автоматически вхожу в систему под тем пользователем, который запустил VNC-сервер (не пробовал под root'ом).

Я хотел бы, чтобы моя система (Slackware 10.2) запускала VNC при загрузке, чтобы я мог видеть экран входа в систему (*KDM*/XDM). У меня сейчас установлен 4-й уровень запуска. Есть идеи, советы или, может быть, предложения по поводу другого ПО? У моего сервера нет монитора.

Psykx, с форума LXF

О Все, что Вам нужно – грамотно настроить VNC-сервер. Замечание: VNC-сервер должен быть запущен и настроен на запуск Вашего любимого оконного менеджера. Вы можете сделать это, отредактировав **$HOME/.vnc/xstartup**. Используйте **startkde&** для запуска KDE, **gnome-session&** – для Gnome, **fvwm2&** – для *FVWM2*. Также убедитесь, что Вы запустили *vncpasswd* в **$HOME/.vnc/passwd** для создания файла паролей.

Red Hat предоставляет простой путь запуска VNC-сервера во время загрузки системы. Используйте **linuxconf** для настройки скрипта *vncserver* (**/etc/init.d/vncserver**) на запуск во время загрузки. Тот скрипт, что есть по умолчанию, не дает, однако, всей гибкости. Откройте **/etc/init.d/vncserver**, отыщите следующую строку:

```
su - ${display##*:} -c \"cd && [ -f .vnc/
passwd ] && vncservers :${display%%:*}\"
```

и замените ее на

```
su - ${display##*:} -c \"cd && [ -f .vnc/
passwd ] && vncservers ${ARGS}
:${display%%:*}\"
```

Затем перейдите к следующему фрагменту:

```
# The VNCSERVERS variable is a list of
# display: user pairs.
# Uncomment the line below to start a VNC
# server on display :1 as my 'myusername'
# (adjust # this to your own).
# You will also need to set a VNC password;
 # run 'man vncpasswd' to see how to do
# that.
# DO NOT RUN THIS SERVICE if your local
# area network is untrusted! For a secure
# way of using VNC, see <URL:http://www.
# uk.research.att.com/vnc/sshvnc.html>.
VNCSERVERS="1:jdimpson"
ARGS="-geometry 1024x768 -alwaysshared"
```

Смените **1024x768** в ARGS на разрешение, соответствующее Вашему дисплею. Можно добавить и другие аргументы в ARGS. Также замените jdimpson на имя того пользователя, от которого Вы хотите запускать VNC.

Значение **1** в VNCSERVERS заставляет запускать VNC на первом дисплее. Вы можете добавить альтернативные дисплеи под именами других пользователей:

```
VNCSERVERS="1:jdimpson 2:phred
3:sysadmin"
```

Удостоверьтесь, что сервер запущен. Если нет, то запустить его можно так (на Red Hat):

```
/etc/init/d/vncserver start
```

Теперь Вы можете соединяться с сервером с помощью любого VNC-клиента. **СЛ**

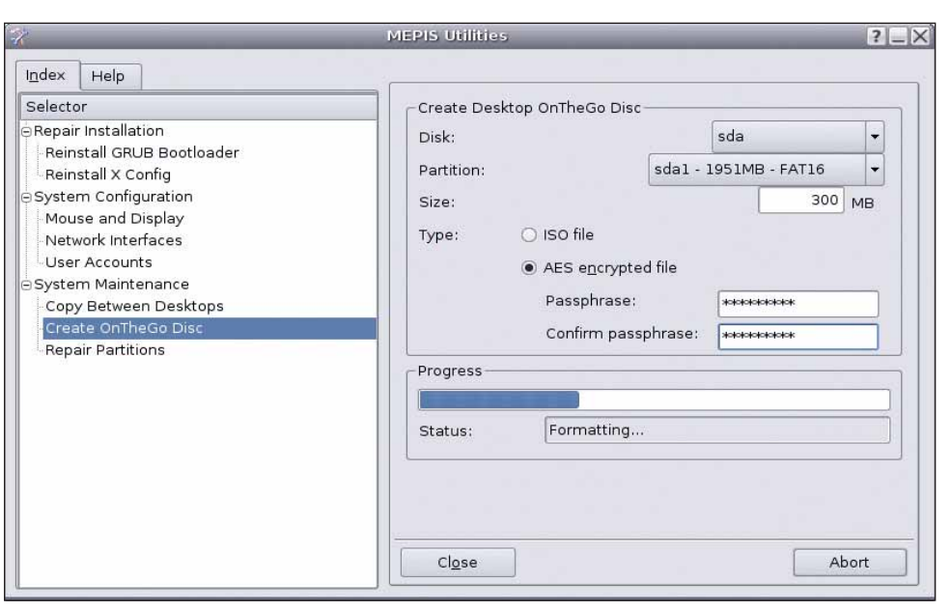
11 Как-то одномерно

В Успешно установил Mandriva Linux 2007 и теперь пытаюсь включить 3D-эффекты. Когда я кликаю на иконке 3D-эффектов в центре управления, все пункты сереют и появляется окно, говорящее, что моя система 3D-эффекты не поддерживает.

У меня GeForce 6600 от Nvidia, она прекрасно работала с Mandriva 2006. Как мне получить трехмерный рабочий стол?

Зэкари [Zachary], с форума LXF

О Скорее всего дело в том, что Вы используете свободный драйвер *nv*. Он не поддерживает 3D-ускорение, тут нужен фирменный драйвер от Nvidia, который можно скачать с www.nvidia.com »



› OnTheGo – мобильная домашняя директория на USB-носителе для LiveCD Mepis.

» в виде инсталляционного пакета. Однако перед установкой драйвера понадобится установить еще кое-какие пакеты – как минимум, исходные тексты ядра. Mandriva больше не включает их в установочные диски, поэтому Вам нужно добавить онлайн-репозитории Mandriva в Центре управления.

Вам также потребуется установить компилятор. Инсталлятор Nvidia поставляется с прекомпилированными модулями для некоторых ядер, а для других модуль собирается в процессе установки. Установив драйверы, Вы должны отредактировать файл конфигурации *X*-сервера, чтоб использовать новый драйвер. Инсталлятор Nvidia требует, чтобы Вы выполняли его без запущенного *X*-сервера, в чистой консоли.

К счастью, есть путь попроще. Penguin Liberation Front (PLF) – это «официальный неофициальный» репозиторий для Mandriva, где имеются несвободные (as in speech) пакеты, которые юридически нельзя включить в дистрибутив (например, *libdvdcss*, необходимый для просмотра зашифрованных DVD). Первый шаг – добавить этот и официальные репозитории в вашу систему. Посетите **http://easyurpmi.zarb.org** и выберите подходящие зеркала Mandriva и PLF. Лучше выбирать ближайшие к вам. Щелкните Proceed, и увидите изрядное количество команд, которые нужно ввести в терминале для выполнения этого шага. Но их можно скопировать. Из меню Mandriva откройте терминал System > Terminals > Konsole, введите **su** для перехода в режим root и выделите мышью команду из браузера.

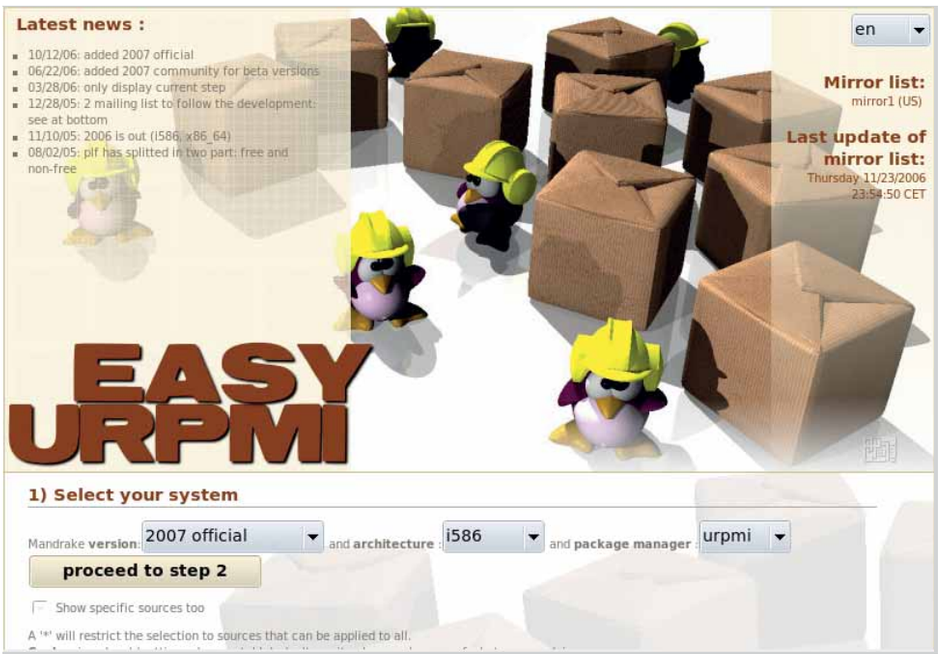
Теперь поместите мышь над терминалом, нажмите среднюю кнопку для вставки команды, а затем – Enter. Если Вы в сети, то через несколько минут будут скачаны списки доступных для установки пакетов.

Теперь запустите Центр управления Mandriva (Система > Настройка > Настройка Вашего компьютера), перейдите в секцию управления пакетами и в поле поиска напечатайте nvidia. Выберите пакет (сейчас это **nvidia-8774-4plf**, но версия может поменяться, когда Вы будете это читать) и нажмите Применить. Если для установки этого пакета потребуются другие, они будут установлены автоматически, то есть выбрать нужно только один пакет.

Наконец, перейдите в секцию Оборудование > Графический сервер и выберите драйвер Nvidia для Вашей видеокарты.

Когда Вы перезагрузитесь, драйверы NVIDIA будут доступны во всем их великолепии, и Вы сможете настроить 3D-эффекты. Наслаждайтесь! **НБ**

› Про EasyURPMI можно прочесть в учебнике Первые шаги на стр. 54.



› **EasyURPMI и вправду прост. Добавляйте репозитории в Mandriva несколькими щелчками мыши.**

12 От @ до “

У меня наблюдается небольшая проблема после установки SUSE 10.1. Пытаюсь ввести @, а получаю “. И наоборот, вместо двойной кавычки появляется «собака». При установке я выбрал английский (UK) язык, и он отображался на экране перед самым началом установки. Надеюсь, есть возможность исправить это без переустановки. Можете помочь?

С форума LXF

О Надо изменить раскладку клавиатуры. Для этого отредактируйте файл **/etc/X11/xorg.conf**, предварительно сделав резервную копию файла, на случай, если вдруг что-то испортите. Эта операция потребует прав root (**su** -). Просто откройте терминал и перейдите в **/etc/X11**.

Для создания резервной копии наберите:

```
cp xorg.conf xorg.conf-back
```

Откройте файл в вашем любимом редакторе. Отыщите строку Option “**XkbLayout**” “язык” и замените язык на **gb**, сохраните файл и перезагрузите машину (**shutdown -r now**). **СЛ**

13 Поболтаем?

В Я настроил сеть между двумя компьютерами под управлением Mandriva Linux 2006, соединив их посредством двух сетевых плат и кабеля. С одной стороны, это смешно (два компьютера в сети), но на деле очень удобно, поскольку раньше для того, что бы передать что-нибудь друг другу, приходилось бегать туда-сюда с «флэшками»... Сейчас для обмена файлами используем Samba, установленную с диска Mandriva. Но возникла проблема: понадобился чат. Я знаю о том, что существуют сетевые чаты для Windows, но о сетевых чатах для Linux я не слышал ничего. Подскажите, пожалуйста, какие программы для Linux позволяют общаться по сети, не используя Интернет (и как их настроить)? Если таких програм нет, подскажите, как сделать что-то вроде ICQ-сервера (чтобы использовать в качестве сетевого чата *Gaim*).

И еще один вопрос: как запускать программу, написанную на Python? Я установил Python, набрал:

```
$ ./mount-image.py
Traceback (most recent call last):
  File “./mount-image.py”, line 3, in ?
    import wx, os, sys, shutil, re
ImportError: No module named wx
```

Я где-то читал, что программы написанные на этом языке, не требуют компиляции, но как их запускать, я не знаю. Подскажите, пожалуйста.

andrewkn

О Чтобы пообщаться с соседом по локальной сети, совершенно необязательно настраивать собственный сервер ICQ или Jabber. Я рекомендую вам обратить внимание на программу *TriX* (**trix.sourceforge.net**). Она хорошо интегрируется с рабочим столом KDE, устанавливаемом в Mandriva Linux по умолчанию и поддерживает как простой обмен текстовыми сообщениями, так и передачу файлов, аватары и прочие «полезности». Кроме того, *TriX* использует тот же протокол, что и популярный *Vypress* Chat для Windows, так что, когда ваша сеть расширится, вы сможете без проблем общаться с товарищами по сегменту, которые еще не перешли на Linux. Единственный минус – *TriX* не входит в состав Mandriva, и вам придется самостоятельно собрать его из исходных текстов.

Что касается второго вопроса, проблема кроется не в способе вызова программы. Оболочке Linux все равно, является ли файл двоичным (откомпилированным) или текстовым (скриптом) – важно, чтобы в его правах доступа был установлен бит ‘**x**’ (eXecutable). Для скриптов, не являющихся исполняемыми файлами (без установленного бита ‘**x**’) возможен альтернативный вариант запуска: интерпретатор имя_скрипта, например: **python mount_image.py**. В данном случае скрипт благополучно запускается на выполнение, но завершается с ошибкой, потому что интерпретатор Python не может найти модуль wx – привязки к Python для библиотеки *wxWidgets* (см. стр. 13). Установите пакет *wxPythonGTK* – все должно заработать. **ВС**

14 Интернет на всех

В У меня сложилась такая проблема. На моем компьютере установлен Mandriva Linux 2007 Free. Данная машина подключена к локальной сети и напрямую к Интернету по технологии ADSL. Вопрос: как можно сделать так, чтобы Интернет с моего компьютера был доступен на всех машинах сети? Остальные машины управляются ОС Windows XP Professional.

Рак А.П.

О В состав Mandriva Linux входит специальный мастер, позволяющий разделить Ваше Интернет-соединение между несколькими компьютерами. Откройте Центр Управления Mandriva и перейдите в раздел Network & Internet. Нужная Вам

опция называется Internet Connection Sharing. Аналогичными возможностями обладает программа *Firestarter* (www.fs-security.com) – своего рода аналог настольного брандмауэра для Windows. Во время первого запуска программы появится мастер, на втором шаге которого следует отметить галочкой пункты «Internet Connection Sharing» и «Enable DHCP for local network», если Вы хотите, чтобы ваши клиенты Windows XP настраивались автоматически (для этого в свойствах их сетевых карт должна быть выбрана опция «Получать IP-адрес автоматически»). Последний способ будет работать не только с Mandriva, но и со всеми популярными дистрибутивами. **BC**

Нужна помощь!

› Для наилучшего ответа на ваш вопрос нам нужно знать как можно больше подробностей. Детально опишите конфигурацию системы. Если вы получили сообщение об ошибке, приведите текст сообщения и точно опишите вызвавшие его действия. Если у вас проблемы с оборудованием, то опишите его. Если Linux уже запущен, то выполните в root-терминале следующие команды и прикрепите к письму файл **system.txt**:

```
uname -a >system.txt
lspci >>system.txt
lspci -vv >>system.txt
```

› Пожалуйста, помните, что сотрудники журнала НЕ являются авторами или разработчиками Linux, любых пакетов или дистрибутивов. Зачастую люди, отвечающие за приложения, выкладывают большую часть информации на web-сайты. Попробуйте почитать документацию!

Мы стараемся ответить на все вопросы. Если вы не нашли ответ на свой, это, возможно, потому, что мы уже ответили на похожий вопрос.

Технология счастья

SUNRADIO.RU

сетевое радио под ключ на базе Linux
новое будущее вашей компании

pr@sunradio.ru +7 812 955 76 70 www.sunradio.ru

Лучшие новинки открытого ПО на планете

LXF HotPicks

В ЭТОТ РАЗ ТОЛЬКО ДЛЯ ВАС: Partition Image » GElemental » Weatherspect » Texmacs » FreeCol » GLChess » Exaile » KAlarm » EDB » Smb4k

Ричард Драммонд
Ричард – свободный разработчик, писатель и отец двух детей. Он живет в Индиане, США, где отчаянно скучает по британскому ТВ, подогретому пиву и сосискам.

Инструмент резервного копирования

Partition Image

LINUX FORMAT HotPicks повторный визит

Версия 0.6.5 **Сайт** www.partimage.org

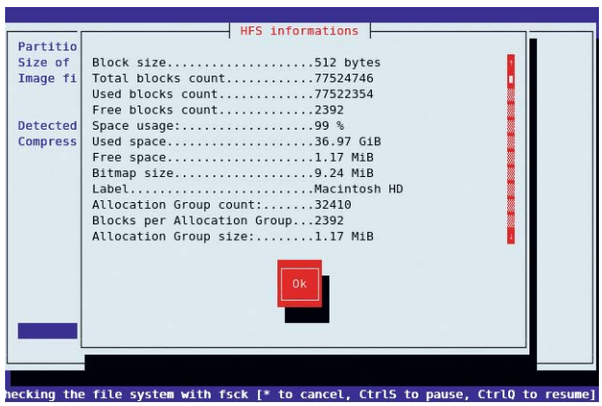
Partition Image – подарок судьбы: инструмент для простого, эффективного создания и восстановления образов файловых систем. Хотите ли вы размножить общий Linux-дистрибутив в сети однотипных компьютеров, или сэкономить время при переустановке ОС, если случилось ужасное – обратите внимание на эту программу.

Да, можно копировать раздел в файл одной простой командой оболочки, но специальный инструмент сделает работу куда быстрее и надежнее. *Partition Image* знает, каким образом разные файловые системы организуют данные на диске, поэтому нет необходимости копировать разделы целиком – достаточно сохранить задействованные дисковые блоки. Это и быстрее, и образ диска получается меньше. Кроме того, *Partition Image* умеет сжимать образы и разбивать их на части фиксированного размера, удобные для размещения на съемных носителях.

Потенциальный недостаток *Partition Image* – необходимость явной поддержки файловых систем, предназначенных для копирования. К счастью, сегодня программа умеет обращаться с разделами ext2, ReiserFS, JFS, XFS и FAT, что покрывает большую часть потребностей Linux, а для HFS+, UFS и NTFS есть экспериментальная поддержка.

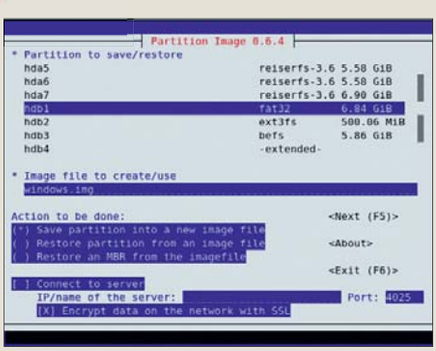
Partition Image имеет дело только с размонтированными файловыми системами, а как же корневой раздел? Решение – использовать отдельный загрузочный диск с установленной *Partition Image*: например, проект предлагает инсталляционный образ, на основе которого можно записать загрузочную дискету или CD-ROM. *Partition Image* может хранить образы на удаленном сервере (возможно шифрование с SSL). Клиент-серверная архитектура *Partition Image* доведена до совершенства в проекте *Clonezilla* (см. http://Clonezilla.sourceforge.net), способном рассылать полноценный дисковый образ по компьютерной сети с помощью односторонней связи.

Последний раз мы рассматривали *Partition Image* два года назад (LXF62). В будущих версиях планируется осуществить поддержку неизвестных файловых систем, использовать алгоритм LZO для быстрого сжатия образов и усовершенствовать поддержку NTFS привлечением кода из проекта **ntfsprogs**. Ждем с нетерпением!



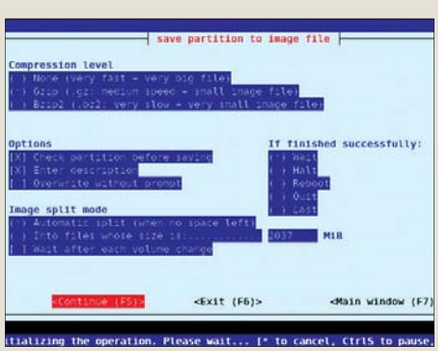
» *Partition Image* работает на машинах PPC, но, к сожалению, пока не справляется с блоками файловой системы HFS+.

Шаг за шагом: Получение дискового образа



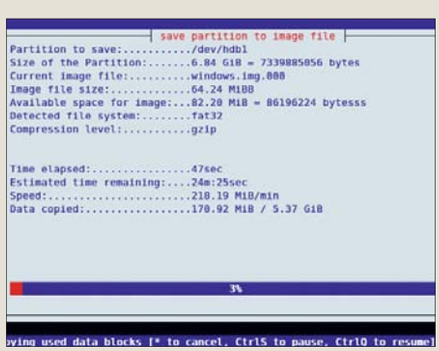
» Выбор цели

Выберите размонтированную систему для сохранения в файле и имя образа. Нажмите F5.



» Настройка параметров

Укажите метод сжатия (рекомендуем *gzip*) и способ разбиения, затем нажмите F5. Добавьте описание и снова нажмите F5.



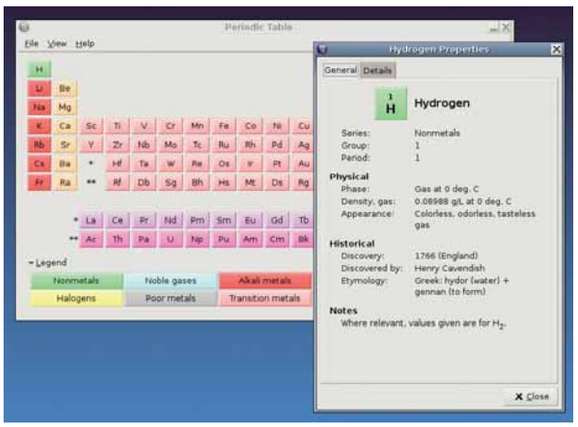
» Запись образа

Partition Image проверит файловую систему и выведет статистику. Выберите кнопку OK и нажмите Enter. Резервирование пошло!

Обозреватель периодической таблицы

GElemental

Версия 0.9.2 **Сайт** www.kdau.com/projects/gelemental



› Все вместе (глубокий вдох): «Литий, натрий, водород, калий, фосфор, кислород…»

В прошлом году ученые-химики объявили, что нестабильный тяжелый элемент под условным названием унунктий – не плод больного воображения и может существовать реально (впрочем, недолго). Действительно ли это прогресс, или ученая братия рассчитывает на процент с продажи обновленных химических талмудов? Не беремся судить, но в любом случае у нас есть альтернатива для нищих студентов-химиков всего мира: *GElemental*, обозреватель периодической таблицы с открытым исходным кодом.

GElemental произошел от *GPeriodic*. Проект был портирован на C++, код реорганизован, графический интерфейс *GTK* модернизирован с помощью привязок gtkmm++, обновлены химические данные и добавлены переводы на другие языки. Все сделано добротно, результат налицо.

Как и его родитель, *GElemental* умеет представлять элементы и в общепринятом виде, как завещал Менделеев, и в виде списка. В первом случае элементы подсвечиваются соответственно их химическим группам или агрегатному состоянию (газ, жидкость, твердое тело) при любой температуре по вашему выбору. Список допускает сортировку по различным критериям: от атомного веса и имени до даты открытия.

Щелкните по элементу в любом представлении; откроется выпадающее окно с более подробным описанием. Здесь помещены общие сведения: физические свойства, исторические факты (дата открытия и имя открывателя) и даже происхождение названия. На другой вкладке приведена более подробная химическая статистика, в том числе атомные, электронные и термодинамические свойства, например, теплоемкость.

GElemental – полезный источник данных, и очень удобный. Мы, правда, заметили отсутствие некоторых данных по сравнению с бумажными учебниками по химии. Почему бы, например, не сортировать элементы по электроотрицательности или удельной теплоемкости, или не добавить еще сведений (например, о стабильности). Но автор рьяно стремится к совершенству, и мы ожидаем появления недостающих функций в следующих версиях.

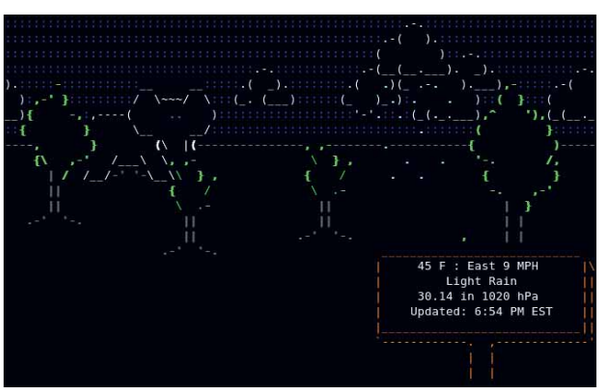
> «Альтернатива для нищих студентов-химиков всего мира.»

Одно из обещанных дополнений для будущих версий – интеграция с Blue Obelisk Data Repository (хранилище данных, разделяемое различными химическими приложениями) – еще один шаг в сторону информативности и достоверности (больше глаз – меньше ошибок). Если это удастся осуществить, успех гарантирован.

Метеоигрушка

Weatherspect

Версия 1.7 **Сайт** http://robobunny.com/projects/weatherspect/htm



› *Weatherspect* – это как выглянуть в окно. Облака, дождь, деревья, вот проходит слон… на севере штата Индиана? Минуточку!

Вы подвержены британской национальной болезни – помешательству на погоде? Вы никогда не пропускаете прогноз для судоходства? Проводите часы, приникнув к метеорологическому каналу? На вашем рабочем столе постоянно обновляются записи о погоде? Если на все вопросы был ответ «да» – мы добыли вам крутой хак, вам понравится.

Weatherspect воспроизводит на терминале анимированные сцены, имитирующие погодные условия, с помощью псевдографики ASCII. Только сообщите, где живете – и программа тут же оценит данные, воспроизведет метеорологически точную картину; в окно больше выглядывать незачем. Например, изображение облачности и цифровые данные о скорости ветра используются для анимации тучек, сыплющих соответствующего вида осадками. Если идет снег, вы увидите снежинки и снеговиков. Если небо чистое, будет нарисовано солнце в правильной позиции днем или луна в соответствующей фазе – ночью.

Честно говоря, погода часто бывает пасмурной, особенно в умеренном климате, перепавшем Британии. *Weatherspect* оживляет картину флорой, фауной и прохожими. Изредка попадается мелочь вроде птичек, кроликов, а то и улиток. Деревья растут, плодоносят и увядают. В небе появляются самолеты, пролетают спутники. Сюрреализм заходит и дальше, но не будем портить сюрприз.

Weatherspect написан на Perl и требует наличия некоторых дополнительных модулей, которых у вас, возможно, еще нет: Term::Animation, для управления псевдографикой ASCII, и **Weather::Underground** для загрузки метеорологических данных. Инсталлировать модули можно с помощью CPAN. От имени суперпользователя запустите инструмент cpan в оболочке и наберите:

```
install Term::Animation
install Weather::Underground
```

Легко. Перед первым запуском *Weatherspect* необходимо настроить. Запустите программу с ключом **-c**, затем просто следуйте инструкциям на мониторе.

> «Скажите, где вы живете. Если идет снег, вы увидите снежинки.»

Честно говоря, надо полагать, что практической ценности у *Weatherspect* никакой, но слишком уж трудно удержаться и не попробовать. Как минимум вы сможете подивиться на хакерское искусство парней, создавших программу, и пожелать им более полезного времяпрепровождения.

Редактор документов

TeXmacs

Версия 1.0.6.7 **Сайт** www.texmacs.org

Когда Линус Торвальдс впервые вдарил по клавишам Vic 20, звезда компьютерных наук Дональд Кнут организовал революцию в цифровом наборе с помощью *TeX*. Кнут остался недоволен качеством печати последнего тома своего многотомного шедевра *The Art of Computer Programming* (Искусство программирования) и написал кое-какое ПО, чтобы исправить дело.

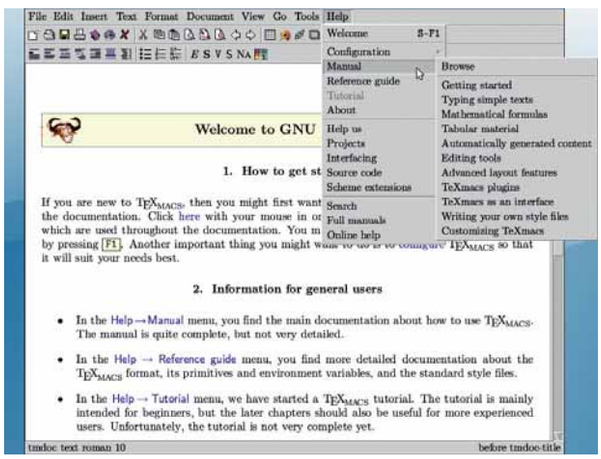
Результат – *TeX*, был принят на ура академической общественностью, так как он превосходно обрабатывал математические формулы. Но так как для этого требуется помечать документы командами *TeX* или, чаще, макросами, а потом вызывать *TeX* для выработки печатного варианта, поколению WYSIWIG программа кажется слишком сложной. Тут выступает на сцену *TeXmacs*, визуальный редактор документов, основанных на правилах набора *TeX*.

TeXmacs можно рассматривать как RAD (среду быстрой разработки) для *TeX*. Как и любая другая среда, она предназначена в общем не для облегчения работы, а для увеличения продуктивности. *TeXmacs* экономит время, давая возможность конструировать и редактировать документы визуально (без разделения на фазы редактирования и визуализации), но изучить его не так-то просто. Конечно, большинство инструментов набора *TeXmacs* достижимо всего несколькими щелчками мыши – или, для мастеров, через несколько клавиатурных комбинаций – но ведь про них еще нужно узнать.

Математика

TeXmacs нацелен на пользователей, которым часто приходится создавать документы с большим количеством научных символов и математических формул. Поэтому здесь есть два основных режима редактирования: текстовый и математический. Интерфейс и кнопки панелей инструментов подстраиваются под текущий режим. В текстовом режиме, например, есть инструменты для смены шрифта, а в математическом предлагаются выпадающие палитры с математическими символами. Там есть все операторы, о которых вам приходилось слышать, и еще кое-что; множество скобок всевозможных фасонов; греческие буквы, каллиграфические и другие специальные символы, которыми обычно выделяют переменные. Инструмент Table (Таблица) добавляет поддержку матриц, определителей и других табличных условных обозначений. И все это носит отпечаток красоты и точности *TeX*.

Чтобы вы не бились над внешним видом вашего творения, *TeXmacs* поддерживает структурированное создание документов с помощью встроенных стилей – включая стили, заимствованные у *LaTeX* (article, book или letter) – наряду с различными классическими стилями научных журналов. Стиль позволяет легко разбить документ на компоненты типа заголовок, глава, главка, абзац, или более специфические части, вроде теоремы, леммы или доказательства. Стиль документа заставит *TeXmacs* следить за соблюдением выбранных типографских правил, предоставляя вам заботу о содержании.



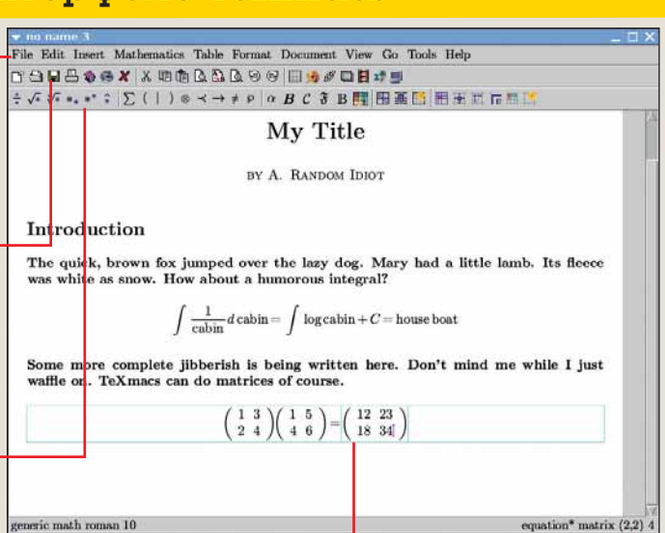
› Освоить *TeXmacs* непросто, но исчерпывающая документация – сработанная, понятно, в *TeXmacs* – поможет «грызть гранит…».

Если понадобились диаграммы Фейнмана, активно применяемые в квантовой теории поля, *TeXmacs* позволит вставить в документы изображения: поддерживаются все стандартные векторные и растровые форматы, а также интеграция с Gnuplot и собственные весьма изощренные инструменты векторного рисования.

После создания документа *TeXmacs* может распечатать его или экспортировать в HTML (формулы обрабатываются либо как изображения, либо при помощи MathML), *LaTeX*, PDF или PostScript. Чтобы восхитить студентов, сопящих у вас за спиной, добавлен модуль презентации, с полноэкранным представлением документа. Поддержка печати, увы, архаична: для отправки документа на принтер нужно указать команду типа lpr или kprinter. Кроме того, мы нашли, что PDF, созданные *TeXmacs*, ставят на колени свободные программы-просмотрщики типа KPDF или Evince (*Adobe Reader* отработал без запинки).

К тому же *TeXmacs* весьма медлителен, а его интерфейс нуждается в обновлении. И тем не менее для *TeX*-новичка это действительно кратчайший путь к созданию математических документов под Linux.

Осваиваем интерфейс TeXmacs



Кнопки для «чего-угодно»
В меню можно найти любую функцию *TeXmacs*, а самым популярным из них присвоена клавиатурная комбинация.

Быстрый доступ
Кнопки основной панели инструментов для: загрузки, сохранения и распечатки документа; откатов, отмены откатов и пр.; вставки таблиц, рисунков и уравнений.

Формулы или текст?
В зависимости от текущего режима эта панель управления предоставляет быстрый доступ к функциям форматирования текста, математическим операторам или инструментам рисования.

Окно редактирования
Подсвеченный прямоугольник обрамляет текущий компонент редактирования. Здесь мы видим матричное уравнение.

С одного взгляда
В строке состояния выводятся запросы системы, отображается информация о текущем режиме редактирования, шрифте и т.п.

HotGames Развлекательные приложения

Стратегия

FreeCol

Версия 0.5.3 **Сайт** www.freecol.org

› Глянь-ка, королева Бесс! Управляй ты колониями вот этак, бостонское чаепитие было бы всемирным!

Популярным жанром фантастики становится пересмотр истории, интересующийся вопросом: что было бы, если бы какие-то исторические события не произошли? Если вы любите отвечать на подобные вопросы, *FreeCol* – ваша игра. Вы получите шанс переиграть колонизацию Америки.

Фактически это перенос в мир открытого ПО классической *Colonization*, игры, созданной Верховным стратегом Сидом Майером [Sid Meier]. Игровая логика *Colonization* близка к его сверхуспешной *Civilization*, и *FreeCol* не отстает. Но у *Colonization*, а значит, и *FreeCol*, более узкая задача: игра начинается с открытия Америки, а вы становитесь вице-королем колоний от избранной вами нации – англичан, французов, испанцев или голландцев. Ваша задача – колонизировать Новый Свет, а затем обрести независимость от собственной нации. Компьютер играет за другие колониальные силы, а можно играть с друзьями через Интернет.

Элементы стратегии и распоряжения ресурсами взяты из *Civ*, но здесь нужны более развитые политические навыки. Задача состоит в росте и развитии ваших колоний, и почти сразу придется умиротворять и аборигенов, и родное правительство. В помощь игрокам создан форум на SourceForge, где можно обмениваться сведениями о том, как извлечь максимум работы и товаров из местных индейцев.

FreeCol написана на Java, для ее работы необходима среда Java 1.5 или выше. Инсталлятор – JAR-файл, который можно развернуть командой:

```
java -jar FreeCol-0.5.3-installer.jar
```

в той директории, куда этот файл загружен. Следуйте экранным подсказкам, и игра будет установлена в выбранное вами место, вместе со значками на рабочем столе. Это как раз нехитрое дело; а вот умудриться избежать войн за территории, катастрофического неурожая и нашествия Макдональдсов – посложнее будет.

> «Придется умиротворять и аборигенов, и родное правительство.»

GUI для шахмат

GLChess

Версия 1.0.0 **Сайт** http://glchess.sourceforge.net

› Черт! Следовало бы мне знать, что компьютер умен и не попадется на детский мат!

Шахматы! Игра королей, возникшая в Индии почти две тысячи лет назад, занимала умы компьютерных гениев почти… да, почти 20 лет. Но, хотя для IBM недавнее достижение чемпионского уровня стоило нескольких миллионов долларов, сегодня ваш скромный настольный Linux-компьютер с шахматным движком сделает то же самое бесплатно.

GLChess – графическое оформление для различных шахматных движков, включая Crafty, GNUChess и Phalanx. Фактически, программа может общаться с любым движком, понимающим Chess Engine Communication Protocol или Universal Chess Interface.

Префикс 'GL' в названии – прозрачный намек на то, что *GLChess* воспроизводит объемное изображение игровой доски; но она приятно выглядит и в двух измерениях, благодаря инструментарию Cairo, обеспечивающему независимый от разрешения рендеринг и сглаживание. Помимо приятной внешности, *GLChess* добавляет игре комфорта за счет функции сохранения игры (поддержка формата Portable Game Notation, или PGN) и возможности проследить историю ходов.

Версия 1.0 стабилизирована для включения в игровой пакет предстоящей версии Gnome 2.18. Жертвой при этом пала сетевая составляющая, но можно бросить вызов компьютеру или гостю. Если вы не хотите ждать выхода Gnome, придется установить немало недостающего. Программа написана на Python и *GTK*, поэтому нужны привязки Python с *GTK* и OpenGL, а для 3D также расширения GL для *GTK* (*PyGTKGlExt*). Последние доступны на домашней странице *GLChess*, вместе с пакетами для Debian и Ubuntu. Библиотека Python Imaging ускорит отрисовку текстур. Мы не сможем помочь вам растормошить мозги, а вот несколько часов наедине с *GLChess* – должны.

> «Префикс 'GL' намекает, что GLChess создает 3D-изображение доски.»

Медиа-плейер

Exaile

Версия 0.2.7b2 **Сайт** www.exaile.org

Загрузка альбомных обложек – это круто, но держать в руках настоящий конверт долгоиграющей пластинки – совсем другое дело.

Прекратив детские споры о том, чей рабочий стол лучше, *Exaile* перенес лучшие качества AmaroK с KDE на Gnome.

В наши дни простого воспроизведения музыкальных дорожек недостаточно. Разумеется, *Exaile* умеет это делать с комфортом, управляясь с самыми разными звуковыми форматами (Ogg Vorbis, MP3 и, конечно, старым добрым аудио-CD). Помимо этого, *Exaile* поддерживает сайты интернет-радио *Shoutcast* (при установке инструмента *Streamripper* возможна запись передачи на диск), общественного сетевого сервиса Last.fm и даже ваш iPod. Есть инструменты для автоматической загрузки и отображения альбомной графики текущей дорожки, слов песни и гитарной табулатуры, а в окне встроенного браузера можно просмотреть сведения об исполнителе, взятые из Wikipedia.

Но все эти приятности лишь оттеняют основную функциональность *Exaile*: управление музыкальной коллекцией и создание плейлистов (списков воспроизведения). Музыку можно импортировать откуда угодно, с показом имени исполнителя или названия альбома. *Exaile* хранит сведения о коллекции во встроенной базе данных *SQLite*, поэтому скорость доступа не зависит от размеров музыкальной библиотеки. Плейлисты можно создавать простым перетаскиванием дорожек, но в *Exaile* есть и более мощные инструменты.

Например, можно автоматически создать плейлист из 100 случайно выбранных дорожек или 100 ваших любимых тем. Умная программа позволит составить плейлисты из песен, отвечающих сразу нескольким запросам пользователя – по имени исполнителя, названию альбома, жанру, году выпуска и многим другим критериям. Короче, поиск дорожки для прослушивания больше не проблема – прощай, рытье на полках, забитых аудиодисками.

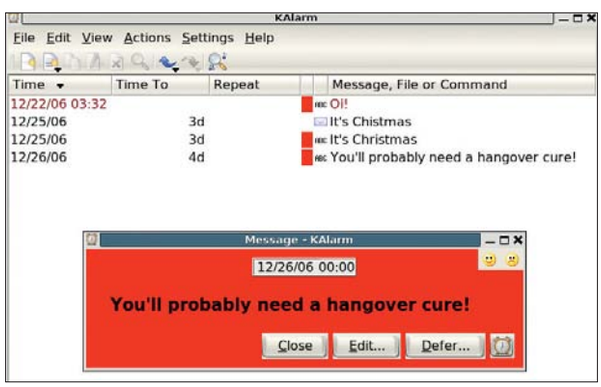
> «Поиск дорожки для прослушивания больше не проблема.»

Как и многие современные приложения Gnome, *Exaile* написан на Python. Вам понадобятся привязки Python к *GTK*, Glade, GStreamer и несколько других библиотек, но, как правило, они уже присутствуют в новейших версиях Gnome. Для поддержки XML *Exaile* требуется также *ElementTree*, включенный в Python 2.5, но если у вас 2.4, установите сами. Загрузите с http://effbot.org/zone/element-index.htm – и слушайте на здоровье!

Планировщик

KAlarm

Версия 1.9.2beta **Сайт** www.astrojar.org.uk/linux/kalarm.html



***KAlarm* способен не только залатать дыры в вашей памяти насчет дат, хотя и тут пригодится!** (надпись в красном диалоговом окне: «Возможен похмельный синдром!»).

Компьютеры придерживаются расписания куда точнее, чем люди. Они никогда не проспят, не опоздают на автобус и не забудут о юбилее. *KAlarm* – одна из программ, обращающих сверхпунктуальность компьютеров в вашу пользу, помогая соблюсти расписание. Программа включена в пакет PIM для KDE, но здесь мы рассмотрим версию еще свежее включенной в последний релиз KDE. Новый *KAlarm* может работать самостоятельно (необходимы лишь *kdelibs*, не весь KDE), а может заменить стандартный экземпляр *KAlarm*.

За простым названием (КБудильник) кроется больше, чем ненавистный звон поутру, хотя *KAlarm* годится и для этого. *KAlarm* обрабатывает событие, к которому можно привязать начало выполнения самых разных задач, не только напоминаний типа «Юбилей любимой тещи!». *KAlarm* может вывести напоминание (дополнительно – произнести его с помощью речевого движка KDE, KTTSD), послать почтовое сообщение, проиграть звуковой файл, запустить заданную пользователем команду или скрипт. События можно запланировать на указанный день, или определенное время дня, или спустя какой-то промежуток времени от указанного часа. Если необходимо, события могут повторяться с регулярными интервалами, или в особые дни, ежегодно и даже при каждом входе в систему. Да, *KAlarm* необычайно гибок – прямо *Cron*, стандартный планировщик Linux, только для людей.

Неудивительно, что *KAlarm* прекрасно совместим с KDE и другими частями PIM, типа *KOrganizer*. Например, можно добавить срабатывание будильника в календарь *KOrganizer* в качестве события. *KAlarm* хранит историю оповещений в стандартном формате iCalendar и запросто импортирует события из других приложений, понимающих этот формат. Например, существует функция импорта Дней рождения из *KAddressbook*.

> «Хорошо совмещается с KDE и другими приложениями PIM.»

Эта тестовая версия очень отличается от последней стандартной. Переработан интерфейс, особенно диалоговое окно редактирования напоминаний; добавлена функция слежения за другими часовыми поясами; исправлено много ошибок, включая обработку високосных годов и переход с летнего на зимнее время; добавлены переводы на другие языки и обновлены прежние. Программа очень полезная, попробуйте обязательно.

Отладчик

EDB

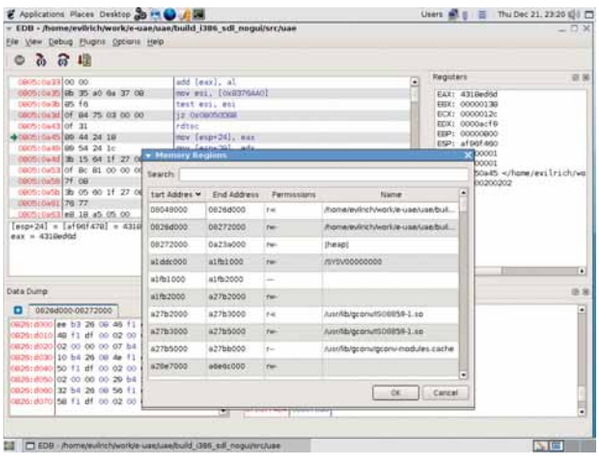
Версия 0.8.11 **Сайт** www.codef00.com/projects.php

Не путайте с покойной звездой хип-хопа (да помнят ли читатели Linux Format ODB?): *EDB* (сокращение от Evan's Debugger, по имени автора, Эван Теран [Evan Teran]) – графический отладчик, написанный *Qt 4*. Это не красивая обертка для некоторых известных отладчиков типа GDB: *EDB* – полностью самостоятельный проект.

С виду он работает так же, как большинство других отладчиков – предоставляет инструменты для установки точек прерывания, просмотра регистров и т.д.; отличия появляются в обработке программно-библиотечных символов. Странно, но их необходимо сбрасывать в отдельный файл с помощью специального скрипта. *EDB* будет обращаться к нему при загрузке соответствующего двоичного файла или библиотеки – такое поведение, вероятно, связано с тем, что *EDB* был создан для анализа программ с закрытым исходным кодом.

Есть мощные инструменты для опроса областей памяти, вероятно используемых приложением, включая стек, кучу и области, занятые подключаемыми библиотеками. Можно просматривать строки и выискивать ссылки на особые адреса памяти, последовательности байтов или группы машинных команд. Эти инструменты уникальны для Linux-отладчиков, и делают *EDB* ценным приобретением для поиска брешей в закрытом коде или инженерного анализа.

› *EDB* незаменим для анализа программ без исходного кода.



Браузер общих ресурсов SMB

Smb4k

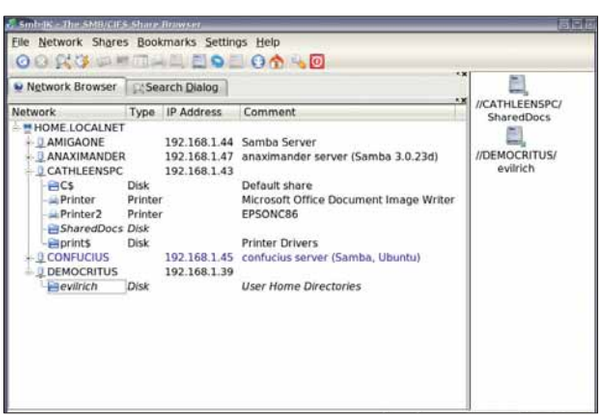
Версия 0.8.0 **Сайт** http://smb4k.berlios.de

Протокол Server Message Block (SMB), основа Windows-сетей, стал de facto стандартом доступа к разделяемым сетевым ресурсам, благодаря открытой реализации, *Samba*. KDE и Gnome, например, имеют встроенную базовую поддержку Samba. Если вашему рабочему столу недостает соответствующих функций, на помощь придет специальный SMB-браузер, *Smb4k*.

У KDE и у Gnome есть общий недостаток: доступ к общим ресурсам по SMB у них возможен лишь через KDE или Gnome, соответственно. А *Smb4k* монтирует общие ресурсы в локальную файловую систему, и к ним может обратиться любое приложение. Он также позволяет настроить и более сложные функции Samba. Идентификация может проходить либо со стандартным паролем, либо со специально назначенным для каждой обменной сессии. Среди других параметров сессии – имя пользователя и групповой ID, с которым монтируется общий ресурс.

Smb4k дает возможность открывать сетевые ресурсы в *Konqueror* одним щелчком мыши и предоставляет диалоговое окно для отправки файла прямо на сетевой принтер. Эти глубоко продуманные решения, плюс более тонкие инструменты – например, способность синхронизировать содержимое локальной папки с сетевым ресурсом через rsync – делают *Smb4k* незаменимым средством работы в сети.

› *Smb4k* сканирует сеть в поисках SMB-ресурсов и находит известные хосты по имени или IP-адресу.



Также выпущены

Новые и обновленные программы, заслуживающие вашего внимания...

Battle Just Started 0.0.5 Танковое сражение на базе OpenGL.
http://bjs.sourceforge.net

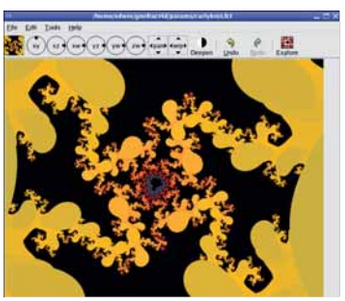
Darkhttpd 1.5 Быстрый, легкий web-сервер для статического контента.
http://dmr.ath.cx/net/darkhttpd

Desktop Data Manager 0.8.5 Менеджер буфера обмена для Gnome.
http://data-manager.sourceforge.net

GNU Classpath 0.93 Свободная реализация стандартной библиотеки классов Java.
www.gnu.org/software/classpath

Gnofract 4D 3.2 Экспериментальный браузер Mandelbrot и Julia.
http://gnofract4d.sourceforge.net

ISO Master 0.6 Графический редактор ISO-образов. http://littlesvr.ca/isomaster



› Фракталы как 4D-объекты!

Lives 0.9.8.2 Мощный пакет для редактирования видео.
http://lives.sourceforge.net

Pascal Server Pages 1.6.0.0 Web-разработка на Pascal?
http://psp.furtopia.org

Qalculate! 0.9.5 Настольный калькулятор.
http://qalculate.sourceforge.net

RecordMyDesktop 0.3.1 Захват видео с экрана монитора.
http://recordmydesktop.sourceforge.net

SynchroEdit 0.5 Позволяет редактировать документ нескольким пользователям одновременно.
www.synchroedit.com

Trac 0.10.3 Менеджмент web-проектов и отслеживание ошибок.

› Свободно с *Classpath*.

LXF DVD90

В этом месяце: два Live-дистрибутива, два BSD и подборка статей журнала в формате PDF...

Майк Сондерс любовно подбирает содержимое диска *Linux Format*, а также поддерживает сайт **www.linuxformat.co.uk**.

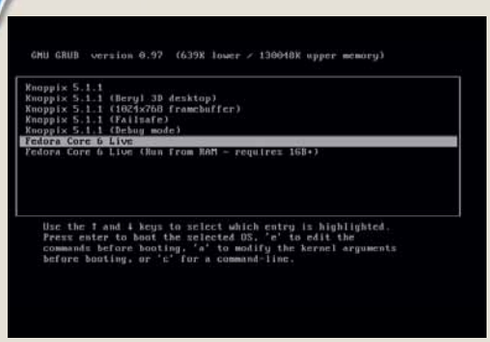
Магия на марше

Linux жив! И вовсе не в виде орд жутких роботов-убийц – пингвинов, сметающих все на своем пути, а в формате Live-дистрибутивов. Knoppix первым запустил и начал популяризировать Live Linux – т.е. версию ОС, которая работает непосредственно с CD и не требует установки на жесткий диск. Когда в 2003 г. появился Knoppix, его повсеместно сочли новшеством, но слишком медлительным для использования в качестве повседневной системы Linux (поскольку CD и DVD работали куда медленнее жесткого диска). Однако многие из современных Live-дистрибутивов работают исключительно в ОЗУ, гораздо быстрее, чем Linux, установленный на жесткий диск; и каждый день они находят все новые и новые сферы применения. Например, Live-дистрибутивы идеальны для тестирования совместимости вашего компьютера с Linux: не требуется проходить через весь процесс установки, чтобы узнать, имеется ли полная поддержка вашего оборудования. А еще они идеальны для демонстрации возможностей Linux потенциальным пользователям: можно просто дать человеку диск, и пусть с ним поработает, не боясь случайно повредить раздел Windows. Но, возможно, самое важное – Live-дистрибутивы открывают дверь в мир новой компьютерной техники: портативную среду рабочего стола. Записав все нужные настройки и файлы на запоминающее USB-устройство, вы можете создать привычную для вас среду почти на любом ПК.

Итак, в этом месяце мы рады предложить вам два лучших Live-дистрибутива на планете: Knoppix, золотого ветерана, и только что выпущенную Live-версию Fedora. Но это только глазурь поверх тортика – вы найдете еще и NetBSD, и OpenBSD, и новый релиз *OpenOffice.org 2.1*, исчерпывающий мастер-класс *Inkscape*, нашу подборку лучших игр и новых релизов программ с открытым кодом. Наслаждайтесь!

mike.saunders@futurenet.co.uk

Шаг за шагом: Запускаем Fedora Core 6 Live

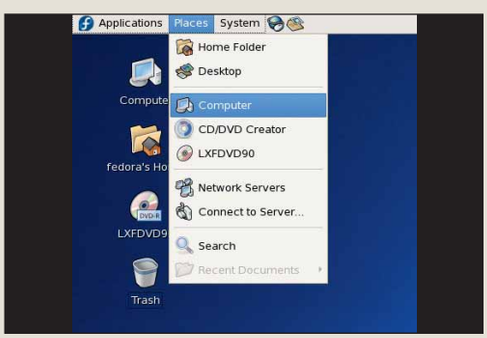


1 Загрузитесь

Загрузите ваш ПК с **LXF**DVD, выберите Fedora Core 6 Live и нажмите Enter. Если у вас более 1 ГБ ОЗУ, выберите опцию ниже для ускоренной загрузки.

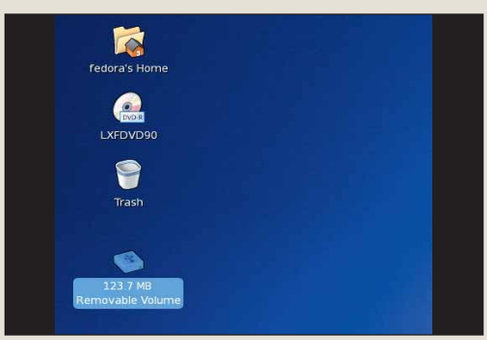
2 Проверьте детали

Загружаясь, Fedora будет показывать на экране информацию о результатах и о ходе загрузки. Можете проигнорировать эту информацию, но если процесс загрузки внезапно прекратится, обратите внимание на отчет об ошибках.



5 Places

Меню Places позволяет вам просматривать информацию о вашем компьютере или получать доступ к другим дискам (например, к DVD-ROM). Через это меню можно также соединяться с другими компьютерами, если вы подключены к сети.



6 Сохранение

Fedora Live грузится с DVD, предназначенного только для чтения, поэтому надо сохранять все создаваемые вами файлы на другом устройстве. Вставьте, например, USB-брелок, и на экране появится соответствующий значок.

Дистрибутив Linux

Fedora Core 6 Live

Два выпуска назад, в LXF87/88, мы включили в наш диск полную версию Fedora Core 6. Исторически, этот дистрибутив, появившийся благодаря Red Hat, существовал только для установки на жесткий диск, но команда разработчиков Fedora заботилась и о Live-версии, и в этом месяце мы рады представить вам плоды их труда. Fedora 6 Live вобрала самые передовые программы Fedora, объединив их в один загружаемый дистрибутив, который работает прямо с DVD, не требуя установки на жесткий диск. А значит, он весьма удобен для проверки совместимости ПК с Linux перед возможной покупкой, для демонстрации Linux друзьям и коллегам, и, наконец, просто ради удовольствия получить новые программы.

› **Fedora – не только работа, но и игра: катим с горки в *Planet Penguin Racer*.**

Fedora Core 6 Live – это Linux, который всегда с вами. Например, вы пришли в гости к другу, и не в силах ни минуты больше стерпеть вида Windows – поэтому берете DVD, вставляете его, и через минуту перед вами знакомый рабочий стол Linux вместо Синего экрана смерти. Fedora 6 Live неотличима от любой версии Fedora; но, естественно, поскольку она работает непосредственно с DVD, по умолчанию на жестком диске не будет ничего сохраняться. Поэтому при перезагрузке все ваши файлы и настройки будут потеряны, если не сохранить их на USB-устройстве или другом носителе.

Чтобы запустить Fedora Core 6 Live, следуйте приведенной пошаговой инструкции. Если хотите продемонстрировать заинтересованному пользователю силу и стабильность Linux, можете использовать это руководство, чтобы показать, насколько просто начать работу в Linux. Требования к оборудованию – минимум 512 МБ ОЗУ и процессор 1 ГГц –достаточны для приемлемой работы системы, хотя дистрибутив, работающий с CD, никогда не сравнится по скорости с дистрибутивом, работающим непосредственно с жесткого диска. Помните, что вы можете создать CD-версию Fedora Live с нашего DVD, следуя инструкциям в **index.html** на DVD. Если при загрузке или при работе с Fedora у вас возникли проблемы, вам всегда помогут на наших форумах на www.linuxforum.ru.

› **Считаете стандартную синюю тему скучноватой? Оживите рабочий стол: пусть он будет в пурпурный цветочек.**

»

3 Gnome

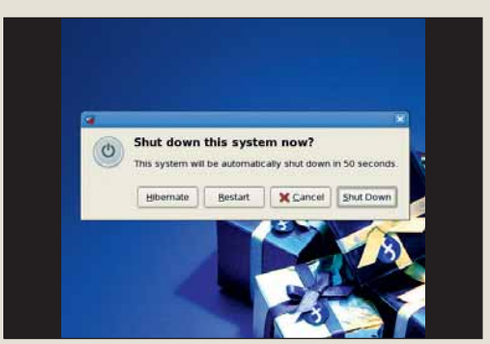
Загрузившись, Fedora запустит рабочий стол Gnome, готовый к работе. Внизу экрана вы увидите панель задач и переключатель виртуальных рабочих столов.

4 Смотрим программы

Откройте меню Applications сверху справа, чтобы увидеть, какие программы включены в Fedora Live. Это офисные инструменты, программы для Интернета и работы с графикой.

7 Настройки

Если вы хотите изменить какие-либо настройки, например, разрешение экрана или раскладку клавиатуры, откройте меню System и просмотрите подпункты Preferences и Administration.



8 Финиш

Закончив работу с Fedora, нажмите меню System > Shut Down, и сможете перезагрузить или выключить компьютер. Не забудьте извлечь DVD перед следующим запуском ПК.

Дистрибутив Linux

Knoppix 5.1.1

Хотя за последние годы появились сотни Live-дистрибутивов, имя Knoppix по-прежнему остается синонимом портативного Linux – и по праву: ведь из всех Live-дистрибутивов это, несомненно, по-прежнему лучший. Основанный на Debian, Knoppix сочетает потрясающее по качеству обнаружение оборудования с богатейшим набором программ, и все это на одном-единственном диске. Просто установите DVD в привод, загрузите компьютер – и перед вами портативная Linux-система, которую вы можете взять с собой куда угодно, она не требует установки на жесткий диск. Хотите показать друзьям возможности Linux – установите DVD в их компьютер, и через пару минут Linux загрузится и заработает.

Даже если вы вполне счастливы с своим дистрибутивом, в этом релизе Knoppix есть функция, которую вы наверняка не упустите – трехмерный рабочий стол *Beryl*, ответвление *Compiz*, принесшее на рабочий стол Linux знаменитые 3D-эффекты и прозрачность, на фоне которых Mac OS X выглядит как старая развалина Windows 3.1. Возможно, вы уже видели рабочий стол-куб и выбор окон а-ля Expose в предыдущих выпусках *Linux Format* – *Beryl* зашел дальше и предлагает такие функции, как просвечивание и отличный менеджер тем рабочего стола. Известно, что и *Compiz*, и *Beryl* весьма сложны в установке, и если они у вас не работают, немедля пробуйте Knoppix. А потом покажите его тем, кто помчался было закупать Vista.

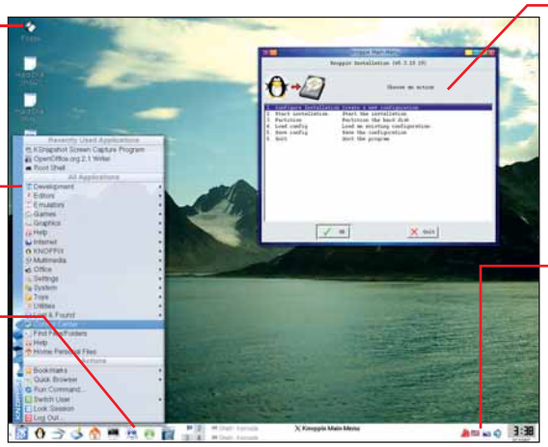
› Из-за проблем с товарными знаками, *Firefox* в Knoppix называется *IceWeasel*.

Knoppix требует минимум 256 МБ ОЗУ и процессор не ниже 1 ГГц, чтобы система работала ровно. Для ее запуска загрузите ПК с **LXF**DVD и нажмите Enter в меню загрузки. Если хотите посмотреть на вышеупомянутые 3D-эффекты рабочего стола, выберите Knoppix 5.1.1 (*Beryl* 3D Desktop) в меню загрузки, это включит *Beryl*, когда загрузится рабочий стол. Если при загрузке возникли проблемы, можете воспользоваться режимами Framebuffer, Failsafe и Debug, но при этом *Beryl* не будет задействован.

Knoppix можно также установить и на жесткий диск; программа установки считается экспериментальной, но если вы пару раз устанавливали Linux раньше, то проблем у вас возникнуть не должно, просто следуйте подсказкам. Зайдите в меню K > Knoppix > Root Shell, и наберите **knoppix-installer** (требуется минимум 2 ГБ на жестком диске). После установки вы получите систему Debian Unstable, которую можно обновлять из обычных репозиториев Debian.

Еще несколько слов об этом релизе Knoppix: он включает в себя KDE 3.5.5, *OpenOffice.org 2.1* и драйвер NTFS-3G, который позволяет монтировать NTFS-разделы Windows на чтение и запись.

Исследуем функции Knoppix



Главное меню
Для доступа к изобилию приложений Knoppix, нажмите на меню K.

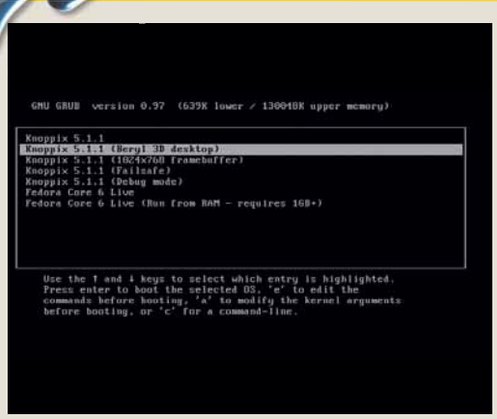
Главное меню
Для доступа к изобилию приложений Knoppix, нажмите на меню K.

Панель
Панель содержит кнопки запуска терминала, просмотрщика файлов *Konqueror* и *OOo*.

Программа установки
Программа установки Knoppix на жесткий диск – для продвинутых пользователей – позволит завести Knoppix на постоянной основе.

Системный лоток
Используя его значки, можно изменить раскладку клавиатуры и подредактировать настройки трехмерного рабочего стола.

Шаг за шагом: Работаем с 3D-рабочим столом Beryl



1 Загрузка
Загрузите ПК с DVD и выберите Knoppix 5.1.1 (Beryl 3D Desktop) из меню загрузки. Linux загрузится с DVD, и через пару минуту вы войдете в рабочий стол KDE.

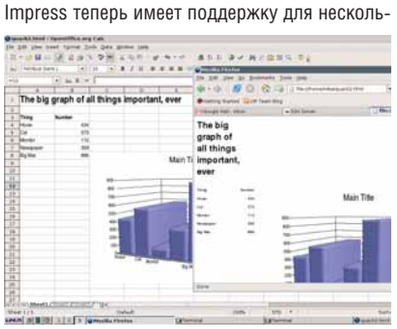
2 Вращение
Нажмите и удержите Ctrl+Alt, затем щелкните левой кнопкой мыши и тащите мышь через весь экран, чтобы полюбоваться эффектом модного рабочего стола в виде куба. Откройте пару окон и переместите мышь вверх-вправо экрана – увидите мини-изображения.

Офисный пакет

OpenOffice.org 2.1

Начать с хорошей новости или с плохой? Начнем с хорошей: *OpenOffice.org* стал чуть быстрее. Ну, еще не совсем в той же лиге скорости, что и *AbiWord*, но в новом релизе 2.1 слегка улучшена производительность. Если вы постоянно пользуетесь *OOo*, то, возможно, заметите одну-две секунды, сэкономленные при загрузке. А плохая новость – процедура установки по-прежнему невероятно глупа. Вы думали, что программа установки у одного из крупнейших и популярнейших открытых проектов, наверное, достойная, так ведь? Как бы не так – у *OOo* нет ни RPMs, ни Debs, ни автопакетов [имеется в виду оригинальная сборка, – прим. ред.], ни даже толковых имен директорий. Мы прилагаем руководство по установке, оно поможет вам продраться сквозь этот нонсенс, установить *OOo* и заставить его работать за несколько минут.

Но сначала – пробежимся по основным функциям *OOo 2.1*. Расширен экспорт HTML в *Calc*, редакторе электронных таблиц, что позволяет более эффективно использовать таблицы стилей, и web-страницы стали больше похожи на исходные электронные таблицы. Инструмент создания презентаций Impress теперь имеет поддержку для нескольких мониторов одновременно, а *Base* обладает куда лучшей совместимостью с *Microsoft Office*. Плюс к тому вышеупомянутое увеличение скорости и сотни исправлений ошибок и прочих усовершенствований по сравнению с версией 2.0.



Экспорт электроннных таблиц в HTML еще не идеален, но постоянно улучшается.

Для установки скопируйте файл **openoffice.org-2.1.tar.gz** из раздела **Office/OpenOffice.org** нашего DVD на жесткий диск. Откройте терминал (например, Gnome Terminal или Konsole) и введите:

```
tar xfvz openoffice.org-2.1.tar.gz
cd openoffice.org-2.1/RPMS/
```

Распакуется архив **openoffice.org-2.1.tar.gz**, и вы перейдете в получившуюся в результате распаковки директорию. (Учтите, мы ее переименовали, поскольку в исходной версии *OOo* ее название нечеловечески длинно!). Перед установкой лучше удалить все старые версии *OpenOffice.org* с помощью менеджера пакетов.

Далее, в системах, основанных на RPM – например, в Fedora, Mandriva, PCLinuxOS или OpenSUSE, вводите:

```
su -c "rpm -ivh *"
```

Команда *su* попросит вас ввести пароль администратора и затем установит нужные пакеты. Если вы работаете в Ubuntu, то ввести вам надо следующее:

```
sudo alien *
sudo dpkg -i *.deb
```

RPM-пакеты конвертируются в Debs, а затем установятся.

Теперь у вас есть *OOo*2.1, расположенный в **/opt/openoffice.org2.1**, и вы можете запустить его из терминала, введя:

```
/opt/openoffice.org2.1/program/soffice
```

Итак, вперед - не забудьте только добавить пунктик в меню или ярлык на рабочий стол!

Удобно, что интерфейс *OpenOffice.org 2.1* почти не изменился; на рисунке показана тема *GTK*.

3 Настройка

Нажмите на красный значок Emerald в системном лотке, чтобы открыть Beryl Settings Manager и настроить эффекты (включая чудесный эффект прозрачности рабочего стола в форме куба).

Документация

Учебники по Inkscape

Несколько наших DVD фокусировались на продвинутом общении с Linux, интересном для крутых пользователей и хакеров – оборудование, программирование и сети. Но в этом месяце наши архивы в формате PDF содержат увесистую коллекцию руководств по *Inkscape*, редактору векторной графики с открытым исходным кодом. Мы все знакомы с растровыми редакторами, типа *Gimp*, но если вам нужны изображения с варьируемым размером, не превращающиеся при увеличении в шахматную доску, то без векторной графики не обойтись (она использует координаты объекта, а не пиксели).

Inkscape вы найдете в разделе **Графика** – это релиз 0.45, но пусть вас не пугает небольшой номер версии – *Inkscape* уже обладает весьма внушительными способностями и очень надежен, так что его уже на данной стадии можно считать версией 1.0. Однако разработчики хотят снабдить его еще кое-какими функциями до выпуска первого полноценного релиза. Возможно, вам удастся получить свежайшую версию *Inkscape* из репозиториев пакетов, ну, а если нет, скомпилируйте релиз 0.45 из исходного кода на нашем DVD.

Руководство по *Inkscape* – настоящий университет векторной графики.

У нас здесь руководство по *Inkscape* на 45 страницах, написанное его гуру Дмитрием Кирсановым для LXF74–85, в разделе **Журнал/PDFs**. Руководство начнет с самых основ (что такое градиенты и контуры) и доведет вас до работы с текстом и инструментами клонирования. Овладев набором функций *Inkscape*, вы сможете испытать ваши новые умения в парочке проектов – например, создать визитную карточку или сайт. *Inkscape* – прекрасная программа и для начинающих, и для опытных дизайнеров векторной графики; с помощью нашего руководства вы мигом научитесь создавать профессиональные изображения.

Операционные системы

NetBSD, OpenBSD, FreeDOS

Если вам как-то неуютно в Linux, почему бы не испробовать другие замечательные операционные системы с открытым кодом? Мы уже рассказывали о NetBSD и OpenBSD на страницах *Linux Format*, а теперь на нашем диске – самые свежие версии.

Пусть Linux – большая любовь мира x86 ПК, но NetBSD и OpenBSD выдержали славную борьбу, доказав, что они имеют право на жизнь. Например, NetBSD включает многие программы, используемые в Linux (в том числе, KDE, *Firefox* и множество других), но при этом обладает преимуществом переносимости исходного кода: он работает более чем на 50 архитектурах, вплоть до экзотических Sega Dreamcasts и карманных компьютеров HP – и при этом имеет единое дерево исходного кода (в отличие от Linux, где многие не-ПК архитектуры работают как ответвления). Помимо этого, NetBSD очень компактна и нетребовательна, что превращает ее в идеальную ОС для стареньких 486-х.

А вот OpenBSD гордится своей суперналежной защитой, по праву хвастаясь тем, что за добрых десять лет работы была обнаружена всего одна уязвимость в установке по умолчанию. Именно так – всего одна удаленно доступная брешь (т.е. ошибка в системе безопасности, допускающая проникновения) за целых десять лет. Установка по умолчанию OpenBSD минимальна и сильно заблокирована, но все равно это большое достижение, а команда OpenBSD создала многие инструменты безопасности, используемые в Linux, включая OpenSSH.

Учтите, что OpenBSD и NetBSD – продвинутые ОС Unix: если вы делаете первые робкие шажки Linux, они покажутся вам слишком сложными. Но если вы пользуетесь Linux уже несколько месяцев и решили испытать нечто новое, то попробуйте их – и вы многое узнаете о различных прелестях Unix, приобретете опыт администрирования и просто получите удовольствие от знакомства с новыми ОС.

Вы найдете образы CD ISO для разных платформ в разделе **Дистрибутивы/NetBSD** нашего DVD: x86 ПК, AMD64, серверы SPARC64 и PowerPC Macs. Можете записать эти образы на CD-R и загрузить их на ваш компьютер, следуя инструкциям в документации, содержащейся в той же директории, что и ISO (**INSTALL-i386.txt**). То же касается и OpenBSD (**Дистрибутивы/OpenBSD**), но сюда мы включили только образ x86 ПК. Руководство по полной установке OpenBSD находится в **INSTALL.TXT**.

Эмулируйте

Есть альтернативная опция для ОС, не предусматривающая записи и загрузки с CD-R: использовать эмулятор *Qemu* PC (раздел **Система** нашего DVD) для создания виртуального ПК, где можно опробовать эти ОС. Эта операция выполняется в три шага. Первый – установка *Qemu* с нашего DVD или репозиториев пакетов вашего дистрибутива. Второй – создание образа виртуального жесткого диска, это делается так:

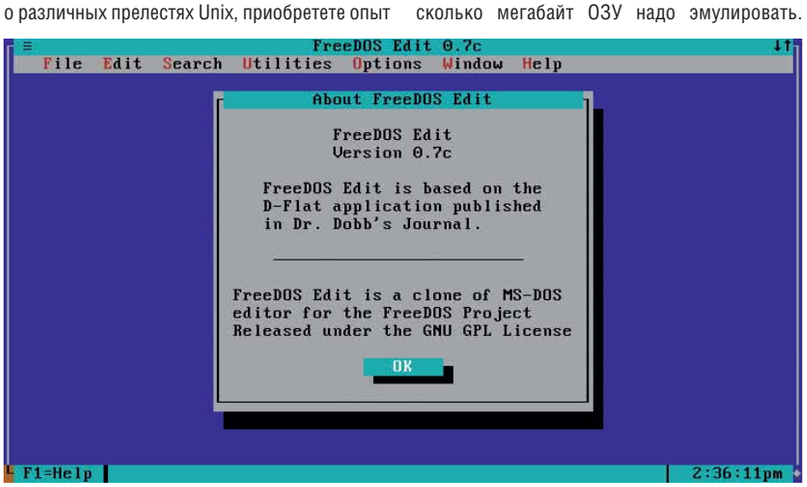
```
qemu-img create virtual-hd.img 3G
```

При этом создается файл с образом виртуального жесткого диска объемом 3 ГБ под названием **virtual-hd.img**.

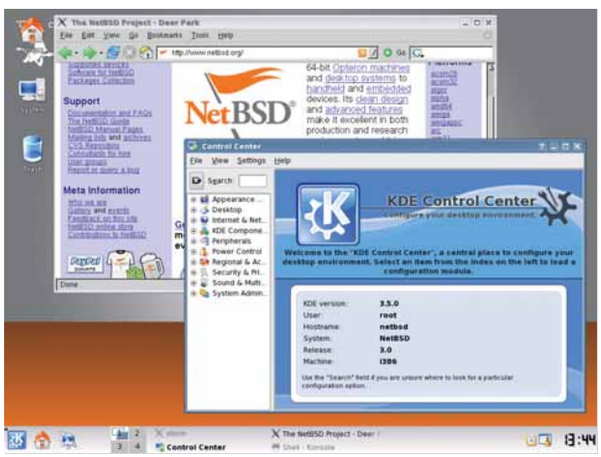
Третий, заключительный шаг – запуск *Qemu* с указанием этого файла виртуального жесткого диска и файла с ISO-образом, который воспроизводит CD-ROM:

```
qemu -cdrom /path/to/netbsd-3.1-i386.iso -
hda virtual-hd.img -boot d -m 256
```

Здесь мы запускаем *Qemu* с виртуального CD-ROM, указывая на наш файл с образом NetBSD ISO и используя файл, созданный ранее в качестве виртуального жесткого диска (**- hda**). Опция **-boot** заставит *Qemu* загрузиться с CD, а опция **-m** в конце сообщит *Qemu*, сколько мегабайт ОЗУ надо эмулировать.



› FreeDOS, клон DOS с открытым кодом, оживляет старину: текстовый режим..



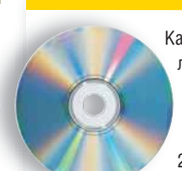
› NetBSD работает на множестве платформ и преспокойно использует KDE, *Apache* и *MySQL*.

Установив ОС на ваш виртуальный ПК, можете удалить опции -cdrom и **-boot** из верхней строки, чтобы загрузиться с жесткого диска.

И в конце – по очереди, но не по значимости – для тех, у кого ностальгия по технологиям прошлых лет: клон DOS с открытым кодом. FreeDOS пригодится, чтобы играть в старые игры под DOS и работать со старыми приложениями, от которых вы не в силах отказаться – вся информация на www.freedos.org.

FreeDOS 1.0 можно установить на эмулятор вышеописанным способом, а можно записать на диск и установить на жесткий диск нормальным способом – образ **freedos-fullws-1.0.iso** находится в разделе директории **Дистрибутивы/FreeDOS** .

И наконец...



Как всегда, предлагаем вам тщательно отобранные, лучшие новые игры для Linux, чтобы скоротать долгие зимние вечера. Самая яркая в нашей подборке – *Warzone 2100*, фантастическая стратегия в реальном времени, изначально выпущенная Eidos Interactive. В 2004, Eidos выпустил игру по лицензии GPL, и на данный момент ею можно пользоваться и распространять совершенно свободно.

Для установки скопируйте файл **warzone2100-2.0.5.package** в домашнюю директорию и запустите его(возможно, сначала вам понадобится сделать его исполняемым) или откройте терминал и введите sh **warzone2100-.0.5.package**. После установки, запустите игру вводом в терминале **warzone2100**, и начните с обучающего режима, чтобы познакомиться с работой игры и с управлением. LXF

› *Warzone 2100*: хватит ли вам ума привести ваши футуристические армии к победе?

ВНИМАТЕЛЬНО ПРОЧТИТЕ ЭТО ПЕРЕД ТЕМ, КАК ИСПОЛЬЗОВАТЬ DVD-ДИСК !

Информация о диске

ЧТО-ТО ПОТЕРЯЛИ?

Часто случается, что новые программы зависят от других программных продуктов, которые могут не входить в текущую версию вашего дистрибутива Linux.

Мы стараемся предоставить вам как можно больше важных вспомогательных файлов. В большинстве случаев, последние версии библиотек и другие пакеты мы включаем в каталог «Essentials» (Главное) на прилагаемом диске. Поэтому, если в вашей системе возникли проблемы с зависимостями, следует заглянуть именно туда.

ФОРМАТЫ ПАКЕТОВ

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или любые другие. Просим вас принять во внимание, что мы ограничены свободным пространством и доступными бинарными выпусками программ. По возможности, мы будем включать исходные тексты для любого пакета, чтобы вы смогли собрать его самостоятельно.

ДОКУМЕНТАЦИЯ

На диске вы сможете найти всю необходимую информацию о том, как устанавливать и использовать некоторые программы. Пожалуйста, не забывайте, что большинство программ поставляются вместе со своей документацией, поэтому дополнительные материалы и файлы находятся в соответствующих директориях.

ЧТО ЭТО ЗА ФАЙЛЫ?

Если вы новичок в Linux, вас может смутить изобилие различных файлов и расширений. Так как мы стараемся собрать как можно больше вариантов пакетов для обеспечения совместимости, в одном каталоге часто находятся два или три файла для различных версий Linux, различных архитектур, исходные тексты и откомпилированные пакеты. Чтобы определить, какой именно файл вам нужен, необходимо обратить внимание на его имя или расширение:

имя_программы-1.0.1.i386.rpm – вероятно, это бинарный пакет RPM, предназначенный для работы на системах x86;

имя_программы-1.0.1.i386.deb – такой же пакет, но уже для Debian;

имя_программы-1.0.1.tar.gz – обычно это исходный код;

имя_программы-1.0.1.tgz – тот же файл, что и выше по списку: «tgz» - это сокращение от «tar.gz»;

имя_программы-1.0.1.tar.bz2 – тот же файл, но сжатый bzip2 вместо обычного gzip;

имя_программы-1.0.1.src.rpm – также исходный код, но поставляемый как RPM-пакет для упрощения процесса установки;

имя_программы-1.0.1.i386.FC4.RPM – бинарный пакет RPM для x86, предназначенный специально для операционной системы Fedora Core 4;

имя_программы-1.0.1.ppc.Suse9.rpm – бинарный пакет RPM, предназначенный специально для операционной системы SUSE 9.x PPC;

имя_программы-devel-1.0.1.i386.rpm – версия для разработчиков.

Если диск не читается...

Это маловероятно, но если все же прилагаемый к журналу диск поврежден, пожалуйста, свяжитесь с нашей службой поддержки по электронной почте: disks@linuxformat.ru

LINUX FORMAT В GIGABYTE DVD

ASP LINUX 11.2

Mono Megapack: Beagle, F-Spot, Blam, Tomboy и другие.
Расширения Firefox: оснастите свой браузер.
Damn Small Linux: старые 486-ые тоже заслуживают любви.
Tor: Анонимный и безопасный Интернет.
Games: Классический Solomon's Key в виде Magicor.

Журнал в формате PDF

Создайте свою Linux-библиотеку вместе с Linux Format!

* Все о сетях
* Linux-графика
* Учебники по OpenOffice.org

openSUSE™

Универсальный дистрибутив для ПК, рабочих станций и серверов
* Ядро 2.6.18.2 * X.Org 7.2 * KDE 3.5.5 * Gnome 2.1.6.1

NEW 10.2 RELEASE

февраль 2007

LXFDVD89

LINUX FORMAT

DVD

Журнал
Blender ..... Модель Tux'a
LinuxFormat ..... Архив прошлых номеров LinuxFormat.
Gtk ..... Код из учебника GTK+
JavaEE ..... Код адресной книги и Jetty 0.6.0rc3
Mono ..... Код из учебника Mono.
PDFs ..... Статьи предыдущих выпусков LXF.
Roundup ..... Программы-трекеры.
Ruby ..... Код из статьи.
Unix API ..... Код примеров статьи.

Рабочий стол
Beagle ..... Поисковый инструмент.
F-Spot ..... Менеджер фотоколлекций.
Kerry ..... Интерфейс KDE к Beagle.
PeaZip ..... Распаковщик архивов.
Tomboy ..... Программа для заметок.
Wyneken ..... Текстовый редактор.
Xpdf ..... Программа просмотра PDF.

Разработка
Medit ..... Текстовый редактор.
Mono ..... Открытая реализация .NET.
MonoDevelop ..... C# IDE.
Nasm ..... Ассемблер x86.

Дистрибутивы
Damn Small Linux ..... Нетребовательный к ресурсам дистрибутив.
SUSE ..... Дистрибутив, спонсируемый Novell.

Игры
Cultivation ..... Социальный симулятор.
Magicor ..... Игра-головоломка.
TuxWordSmith ..... Обучающая игра.

Справка
Rute ..... Руководство по администрированию Linux.

HOTPICKS
Atop ..... Мониторинг процессов.
CuteCom ..... Последовательный терминал.
Dissy ..... Дизассемблер.
Enigma ..... Игра-головоломка.
FLPhoto ..... Программа управления изображениями.
Frozen Bubble ..... Клон Bust-a-Move.
Goggles ..... Интерфейс для DVD-плейера Ogle.
Hardware Lister ..... Сведения о вашем оборудовании.
JMemorize ..... Программа для зубрежки.
NSPluginWrapper ..... Обертка для модулей Netscape.

Интернет
Blam ..... Читалка новостей RSS.
Firefox extensions ..... Дополнения к Firefox.
Tor ..... Анонимайзер.
Tork ..... Контроллер Tor для KDE.

Звук
AudioFormat ..... Конвертор звуковых форматов.
Banshee ..... Плейер и музыкальный менеджер.
Gnomeradio ..... Тюнер FM-радио.
Gnormalize ..... Аудио-конвертор.

Система
D-BUS ..... Система пересылки сообщений.
GXInstall ..... Инсталлятор программ.
Pinger ..... Программа для ping'а нескольких узлов.
Qemu ..... Эмулятор ПК.
Refit ..... Загрузчик для Intel Mac.
Tilda ..... Quake-подобный терминал.

Сервер
Apache ..... Web-сервер.
MyJSQLView ..... Программа просмотра базы данных.
PostgreSQL ..... Сервер баз данных.

Безопасность
FloppyFW ..... Маршрутизатор на дискете.
JPasswordGenerator ..... Генератор паролей.
Sussen ..... Сканер безопасности.

Главное
Avifile ..... Библиотека чтения/записи AVI файлов.
Bash ..... Командная оболочка.
CheckInstall ..... Создатель бинарных пакетов.
Coreutils ..... Утилиты командной строки.
CSV ..... Индекс файлов диска.
GLib ..... Низкоуровневая библиотека C.
Glibc ..... Библиотека GNU C.
GTK ..... Инструментарий пользовательского интерфейса.
HardInfo ..... Информация о системе и тесты.
Jigdo ..... Создатель ISO-образов.
Kernel ..... Свежий релиз ядра Linux.
libsigc ..... Система обратных вызовов для C++.
libXML ..... Анализатор и инструментарий XML.
ncurses ..... Оконный инструментарий текстового режима.
Python ..... Язык программирования.
Rawrite ..... Программа записи образов на дискеты.
SBM ..... Smart Boot Manager.
SDL ..... Библиотека мультимедиа.

СОЗДАНИЕ УСТАНОВОЧНЫХ ДИСКОВ ПРИ ПОМОЩИ CDRECORD

Самый быстрый способ записать ISO-образ на чистую матрицу – это *cdrecord*. Для всех перечисленных ниже действий потребуются права root. Для начала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке, например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле **/etc/default/cdrecord**. Добавьте по одной строке для каждого устройства записи (вероятно, в вашей системе присутствует всего одно такое устройство):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке – это метка, затем, после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь вы можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Все, что вам теперь нужно для записи ISO-образа – это набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не из числа любителей командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее из под root, выберите вкладку Burn и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на Combust!. Пока ваш образ пишется на диск, можете выпить чашечку кофе.

Другая ОС?

Вам не обязательно использовать Linux для записи компакт-диска. Все необходимые файлы уже включены в ISO-образ. Программы вроде *cdrecord* просто переносят данные на чистую матрицу. Если у вас нет устройства для записи дисков, можно найти того, у кого оно есть, и записать диск на его компьютере. На нем может стоять Windows, Mac OS X, AmigaOS, или любая другая ОС.

Нет устройства для записи дисков?

А что, если у вас нет устройства, с помощью которого можно было записать образ на диск? Вы знаете кого-либо с таким устройством? Вам не обязательно использовать Linux для записи дисков, подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт его разработчика. LXF

ВНИМАТЕЛЬНО ПРОЧТИТЕ ЭТО ПЕРЕД ТЕМ, КАК ИСПОЛЬЗОВАТЬ DVD-ДИСК !

Информация о диске

ЧТО-ТО ПОТЕРЯЛИ?

Часто случается, что новые программы зависят от других программных продуктов, которые могут не входить в текущую версию вашего дистрибутива Linux.

Мы стараемся предоставить вам как можно больше важных вспомогательных файлов. В большинстве случаев, последние версии библиотек и другие пакеты мы включаем в каталог «Essentials» (Главное) на прилагаемом диске. Поэтому, если в вашей системе возникли проблемы с зависимостями, следует заглянуть именно туда.

ФОРМАТЫ ПАКЕТОВ

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или любые другие. Просим вас принять во внимание, что мы ограничены свободным пространством и доступными бинарными выпусками программ. По возможности, мы будем включать исходные тексты для любого пакета, чтобы вы смогли собрать его самостоятельно.

ДОКУМЕНТАЦИЯ

На диске вы сможете найти всю необходимую информацию о том, как устанавливать и использовать некоторые программы. Пожалуйста, не забывайте, что большинство программ поставляются вместе со своей документацией, поэтому дополнительные материалы и файлы находятся в соответствующих директориях.

ЧТО ЭТО ЗА ФАЙЛЫ?

Если вы новичок в Linux, вас может смутить изобилие различных файлов и расширений. Так как мы стараемся собрать как можно больше вариантов пакетов для обеспечения совместимости, в одном каталоге часто находятся два или три файла для различных версий Linux, различных архитектур, исходные тексты и откомпилированные пакеты. Чтобы определить, какой именно файл вам нужен, необходимо обратить внимание на его имя или расширение:

имя_программы-1.0.1.i386.rpm – вероятно, это бинарный пакет RPM, предназначенный для работы на системах x86;

имя_программы-1.0.1.i386.deb – такой же пакет, но уже для Debian;

имя_программы-1.0.1.tar.gz – обычно это исходный код;

имя_программы-1.0.1.tgz – тот же файл, что и выше по списку: «tgz» - это сокращение от «tar.gz»;

имя_программы-1.0.1.tar.bz2 – тот же файл, но сжатый bzip2 вместо обычного gzip;

имя_программы-1.0.1.src.rpm – также исходный код, но поставляемый как RPM-пакет для упрощения процесса установки;

имя_программы-1.0.1.i386.FC4.RPM – бинарный пакет RPM для x86, предназначенный специально для операционной системы Fedora Core 4;

имя_программы-1.0.1.ppc.Suse9.rpm – бинарный пакет RPM, предназначенный специально для операционной системы SUSE 9.x PPC;

имя_программы-devel-1.0.1.i386.rpm – версия для разработчиков.

Если диск не читается...

Это маловероятно, но если все же прилагаемый к журналу диск поврежден, пожалуйста, свяжитесь с нашей службой поддержки по электронной почте: disks@linuxformat.ru

Содержание LINUX FORMAT DVD Март 2007

Сторона 1

Рабочий стол
Day_Planner – программа для напоминания о встречах и других событиях
Enlightenment - красивый оконный менеджер с поддержкой графических библиотек
GCstar - свободное приложение для управления вашими коллекциями
GnuCash - программа для персонального учета финансов
Hipo – программа для управления файлами на iPod
KCatalog - утилита для организации каталога фильмов и музыки
TkFlightPlan – программа для планирования полетов
Xpdf - открытый просмотрщик файлов формата PDF

Разработка
FreeBASIC – свободный компилятор языка BASIC
Monket Studio - кроссплатформенная среда разработки приложений
Mono – свободная реализация .NET
MonoDevelop - среда разработки для Mono
wxWidgets – кроссплатформенная библиотека для создания GUI

Дистрибутивы
Fedora Live – LiveCD на основе Fedora Core 6
FreeDOS – свободная версия DOS-совместимой ОС
Knoppix – LiveCD с Xgl и Beryl
NetBSD – переносимая BSD-система
OpenBSD - свободная UNIX-совместимая ОС с упором на безопасность

Игры
Brain Party - двумерный паззл для всей семьи
Enigma - логическая игра, похожая на Oxyd для Atari
Warzone 2100 - стратегия реального времени
X-Moto - двумерный мотосимулятор

Графика
Inkscape - открытый векторный редактор
KPhotoAlbum - утилита для просмотра, комментирования и поиска изображений
Linman - быстрый генератор фракталов Мандельброта

Hot pics
EDB - интерактивный отладчик, подходящий для исследования закрытых программ
Exaile - аудиоплеер, похожий на Amarok, но написанный на Gtk+
FreeCol - свободная версия стратегии Colonization
gElemental - электронная таблица Менделеева
glChess - шахматы для Unix
Kalarm - планировщик и будильник в одном флаконе
Partition_Image - утилита для создания и восстановления образов партиций
Smb4k - утилита для просмотра SMB/CIFS-ресурсов
TeXmacs - свободный текстовый процессор
WeatherSpect - утилита для получения информации о погоде

Интернет
Claws Mail – продвинутый почтовый клиент
Firefox - популярнейший открытый веб-браузер
jPodder - программа для работы с подкастами
Pan - программа для работы с Usenet
WebMonX – утилита для слежения за обновлениями на веб-сайтах
XChat – IRC-клиент

Офис
OpenOffice.org - открытый офисный пакет
Scribus - программа для верстки
SoftMaker 2006 Demo – демо-версия коммерческого офисного пакета

Сервер
Apache – популярнейший http-сервер на планете
Lightbird – сервис для передачи коротких сообщений в локальной сети
mod_cband - модуль Apache2, решающий проблему с лимитом пользователей и скорости
MySQL - популярнейший открый сервер баз данных
SugarCRM - коммерческая CRM с открытым исходным кодом
Tomcat - веб-сервер, написанный на языке Java

Система
GRUB - мультисистемный загрузчик
KThinkBat - апплет для слежение за зарядом батареи
NdisWrapper – инструмент для загрузки NDIS-драйверов
QEMU - открытый эмулятор процессора
QtEmu - графический интерфейс к эмулятору QEMU
TweaK - утилита для редактирования скрытых настроек среды KDE
XenMan - интуитивная графическая утилита для

Сторона 2

FreeBSD 6.2
PC-BSD 1.3

Комментарии? Присылайте ваши мысли и предложения по электронной почте: info@linuxformat.ru
Пожалуйста, ознакомьтесь с опубликованной в журнале инструкцией перед использованием данного диска.
Настоящий диск тщательно тестировался и проверялся на всех стадиях производства, однако, как и в случае с любым новым ПО, мы рекомендуем вам использовать антивирусный сканер. Мы также рекомендуем всегда иметь под рукой актуальную резервную копию данных вашего жесткого диска. К сожалению, редакция Linux Format не может принимать на себя ответственность за любые повреждения, разрушения или иные убытки, которые может повлечь за собой использование этого DVD, представленных на нем программ или данных. Перед тем, как устанавливать какое-либо ПО на компьютер, подключенный к сети, проконсультируйтесь с сетевым администратором.

Дефектные диски. В маловероятном случае обнаружения дефектов на данном диске, пожалуйста, обращайтесь по адресу: disks@linuxformat.ru

Тираж изготовлен на Уральском электронном заводе., 620066, Россия, г. Екатеринбург, ул. Комвузовская 17-203, Лицензия МПТР России ВАФ № 77-15

Поставляется вместе с журналом LINUXFORMAT номер 3(90) Март 2007

DVD

СОЗДАНИЕ УСТАНОВОЧНЫХ ДИСКОВ ПРИ ПОМОЩИ CDRECORD

Самый быстрый способ записать ISO-образ на чистую матрицу – это *cdrecord*. Для всех перечисленных ниже действий потребуются права root. Для начала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке, например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле **/etc/default/cdrecord**. Добавьте по одной строке для каждого устройства записи (вероятно, в вашей системе присутствует всего одно такое устройство):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке – это метка, затем, после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь вы можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Все, что вам теперь нужно для записи ISO-образа – это набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не из числа любителей командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее из-под root, выберите вкладку Burn и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на Combust!. Пока ваш образ пишется на диск, можете выпить чашечку кофе.

Другая ОС?

Вам не обязательно использовать Linux для записи компакт-диска. Все необходимые файлы уже включены в ISO-образ. Программы вроде *cdrecord* просто переносят данные на чистую матрицу. Если у вас нет устройства для записи дисков, можно найти того, у кого оно есть, и записать диск на его компьютере. На нем может стоять Windows, Mac OS X, AmigaOS, или любая другая ОС.

Нет устройства для записи дисков?

А что, если у вас нет устройства, с помощью которого можно было записать образ на диск? Вы знаете кого-либо с таким устройством? Вам не обязательно использовать Linux для записи дисков, подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт его разработчика. LXF

Содержание DVD

ЖУРНАЛ
Blender .....Примеры из статьи.
Gtk .....Примеры из статьи.
Esoteric.....Экзотические языки программирования.
JavaEE.....Примеры из статьи.
Mono.....Примеры из статьи.
PDFs .....Учебники Inkscape.
Roundup.....Клиенты BitTorrent.
PostgreSQL.....Полный текст интервью с Алексеем Харитоновым.

РАБОЧИЙ СТОЛ
Day Planner.....Утилита планирования времени.
Enlightenment...Оконный менеджер.
GCstar.....Менеджер коллекций.
GnuCash .....Персональный финансовый менеджер.
Hipo .....Менеджер данных iPod.
KCatalog .....Инструмент каталога мультимедиа.
TkFlightPlan.....Планировщик.
Xpdf .....Программа просмотра PDF.

РАЗРАБОТКА
FreeBasic.....Компилятор Basic.
Monkey Studio..IDE на базе Qt.
Mono.....Открытая реализация .NET.
MonoDevelop....C# IDE.
wxWidgets .....GUI-инструментарий.

ДИСТРИБУТИВЫ
Fedora .....Live-дистрибутив, выпущенный Red Hat.
FreeDOS.....DOS-совместимая ОС.
Knoppix.....Live-дистрибутив.
NetBSD .....Переносимая ОС.
OpenBSD.....Супербезопасная ОС.

ИГРЫ
Brain Party.....Подборка мини-игр.
Enigma.....Головоломка.
Warzone 2100..Стратегия в реальном времени.
X-Moto.....Гонки на мотоциклах.

ГРАФИКА
Inkscape .....Редактор векторной графики.
KPhotoAlbum....Менеджер изображений.
Linman.....Программа фрактального увеличения.

СПРАВКА
Rute .....Руководство по администрированию Linux.

HOTPICKS
EDB.....Отладчик.
Exaile.....Медиа-плейер.
FreeCol.....Клон Colonization.
GElemental.....Программа просмотра периодической таблицы.
GLChess.....Трехмерные шахматы.
KAlarm.....Персональный планировщик.
Partition Image..Резервное копирование разделов.
Smb4k.....Браузер ресурсов SMB.
TeXmacs.....Платформа для типографского набора.
Weatherspect...Программа просмотра погоды ASCII.

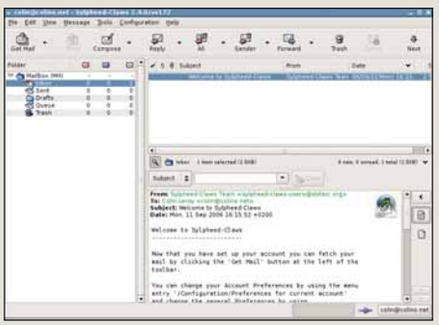
ИНТЕРНЕТ
Claws Mail.....Email-клиент.
Firefox.....Web-браузер.
iPodder.....Получатель подкастов.
Pan.....Программа для чтения новостей.
WebMonx.....Монитор сайтов.
XChat.....IRC-клиент.

ОФИС
OpenOffice.org.....Офисный пакет.
Scribus.....Настольная издательская система.
SoftMaker demo..Пробная версия офисного пакета.

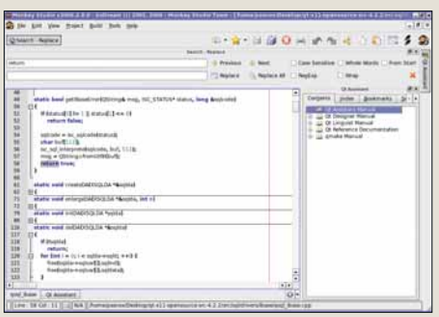
СЕРВЕР
Apache.....Web-сервер.
Lightbird.....IM-сервер и клиент.
mod_cband.....Модуль Apache 2.
MySQL.....Сервер баз данных.
SugarCRM.....CRM-система.

СИСТЕМА
Grub.....Тестовая версия загрузчика.
KThinkBat.....Монитор батареи ThinkPad.
Ndiswrapper.....Слой совместимости с драйвером Windows.
Qemu.....Эмулятор ПК.
QtEmu.....Интерфейс Qt для Qemu.
Tweak.....Менеджер скрытых настроек KDE.
XenMan.....Менеджер виртуализатора Xen.

ГЛАВНОЕ
ATI driver.....Графический драйвер.
Avifile.....Библиотека чтения/записи AVI файлов.
Bash.....Командная оболочка.
CheckInstall.....Программа создания двоичных пакетов.
Coreutils.....Утилиты командной строки.
CSV.....Список файлов, содержащихся на диске.
GLib.....Низкоуровневая библиотека.
Glibc.....Библиотека GNU C.
GTK.....GUI-инструментарий.
HardInfo.....Информация и оценка системы.
Jigdo.....Создатель ISO-образов.
Kernel.....Свежий релиз ядра Linux.
LibXML2.....Анализатор и набор инструментов XML.
Ncurses.....Оконный инструментарий текстового режима.
Nvidia driver.....Графический драйвер.
Python.....Язык программирования.
Rawrite.....Запись образов на дискеты.
SBM.....The Smart Boot Manager.
SDL.....Библиотека мультимедиа.



› У *Claws Mail*, ответвления *Sylpheed*, аккуратный интерфейс в стиле Evolution.



› Ищете новую IDE? В *Monkey Studio* есть подсветка синтаксиса и завершение кода.

› Если вы любите *Kickstart II* со времен 8 бит, испытайте свои виртуальные навыки мотогонщика в X-Moto.

LINUX FORMAT
Главное в мире Linux

Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия
ПИ № ФС77-21973 от 14 сентября 2005 года
Выходит ежемесячно. Тираж 5000 экз.

РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ:
ГЛАВНЫЙ РЕДАКТОР
Валентин Синицын info@linuxformat.ru
Литературные редакторы
Родион Водейко, Елена Толстякова, Иван Мищенко
Переводчики
Александр Бикмеев, Светлана Кривошеина, Александр Кузьменков, Алексей Опарин, Валентин Развозжаев, Сергей Супрунов, Александр Черных, Юлия Шабунио
Допечатная подготовка
Мария Пучкова, Родион Водейко
Креативный директор
Станислав Медведев
Технический директор
Денис Филиппов
Директор по рекламе
Денис Игнатов +7 812 965 7236 advert@linuxformat.ru
Заместитель генерального директора
Софья Виниченко
Генеральный директор
Павел Фролов
УЧРЕДИТЕЛИ
частные лица
ИЗДАТЕЛИ
Станислав Медведев, Виктор Федосеев, Павел Фролов

Отпечатано в типографии «Текст», ООО «ППК «Текст»
188680, Ленинградская область, Всеволожский район, Колтуши, д.32
Заказ _____
Пре–пресс: *d.r.i.v.e-group*

РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ:
Редактор Ник Вейч (Nick Veitch) nick.veitch@futurenet.co.uk
Заместитель редактора Пол Хадсон (Paul Hudson) paul.hudson@futurenet.co.uk
Художественный редактор Эфрейн Эрнандес-Мендоса (Efrain Hernandez-Mendoza) efrain.hernandez-mendoza@futurenet.co.uk
Новостной редактор Майк Сондерс (Mike Saunders) mike.saunders@futurenet.co.uk
Литературный редактор
Ребекка Смолли (Rebecca Smalley) rebecca.smalley@futurenet.co.uk
Штатный автор
Грэм Моррисон (Graham Morrison) graham.morrison@futurenet.co.uk
Ассистент по выпуску
Эндрю Грегори (Andrew Gregory) andrew.gregory@futurenet.co.uk

Авторы
Марк Бейн (Mark Bain), Ладислав Боднар (Ladislav Bodnar), Нейл Ботвик (Neil Bothwick), Энди Ченнел (Andy Channelle), Ричард Драммонд (Richard Drummond), Марко Фиоретти (Marco Fioretti), Стефан Лукас (Stephan Lucas), Евгений Балдин, Андрей Боровский, Андрей Прахов, Петр Семилетов, Александр Супрунов, Алексей Федорчук, Александр Бабаев, Илья Шпаньков
Художественные ассистенты: Зигги Бейкер (Ziggi Baker), Энди Онстед (Andy Ounsted)
Фотографии: Джейсон Каплан (Jason Kaplan)
Иллюстрации: Нейл Бартлет (Neil Bartlett), Пол Блехфорд (Paul Blachford), Elly Walton Illustrations, Крис Винн (Chris Winn)

КОНТАКТНАЯ ИНФОРМАЦИЯ
UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW
Tel 01225 442244 Email: linuxformat@futurenet.co.uk
РОССИЯ:
Санкт-Петербург (редакция): ул. Гончарная, 23, офис 54, телефон: (812) 717-00-37
Представительство в Москве:
пр.Мира, 161, телефон +7(495) 799-18-63, +7(495)136-88-45
Email: info@linuxformat.ru, Web: www.linuxformat.ru

Авторские права: Статьи, переведенные из английского издания Linux Format, являются собственностью или лицензией Future Publishing Ltd (Future plc group company). Все права зарегистрированы. Никакая часть данного журнала не может быть повторно опубликована без письменного разрешения издателя.

Все письма, независимо от способа отправки, считаются предназначенными для публикации, если иное не указано явно. Редакция оставляет за собой право корректировать присланные письма и другие материалы. Редакция Linux Format получает неэксклюзивное право на публикацию и лицензирование всех присланных материалов, если не было оговорено иное. Linux Format стремится оставлять уведомление об авторских правах всюду, где это возможно. Свяжитесь с нами, если мы не упомянули вас как автора предложенных вами материалов и мы постараемся исправить эту ошибку. Редакция Linux Format не несет ответственности за опечатки.

Все присланные материалы могут быть помещены на CD или DVD-диски, поставляемые вместе с журналом, если не было оговорено иное.

Ограничение ответственности: используйте все советы на свой страх и риск. Ни при каких условиях редакция Linux Format не несет ответственность за повреждения или ущерб, нанесенные вашему компьютеру и периферии вследствие использования тех или иных советов.

За содержание рекламных материалов редакция ответственности не несет.

Linux-зарегистрированная торговая марка Линуса Торвальдса (Linus Torvalds). Название «GNU/Linux» заменяется на «Linux» в целях сокращения. Остальные торговые марки являются собственностью их законных владельцев.
Linux Format является торговой маркой Future Publishing Ltd (Future plc group company).

За информацией о журналах, издаваемых Future plc group company, обращайтесь
http://www.futureplc.com

© Linux Format 2005
© Future Publishing Ltd 2005

BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO •SAN FRANCISCO

в апреле

В ожидании Gimp

Новый крупный релиз уже на горизонте, но почему это заняло так много времени, и что произошло с GEGL? Попробуем разобраться...

15 шагов к безопасности

Мы научим вас, как защитить компьютер дома, в серверной и даже в Сети!

Обзор: Nokia N800

Новый интернет-планшет приходит в Россию. Не пора ли запланировать очередную покупку?

Эрик Оллмен

Создатель Sendmail о том, откуда взялся спам.

Внимание! Содержание следующих выпусков может изменяться без уведомления